CLUBE

Z80

Junho/84

N.º 21

Coordenadores: MARIA IRENE • ALEXANDRE SOUSA • J. MAGALHÃES • AV. BOAVISTA, 832-2.º T. • TELEF. 65127 • 4100 PORTO

NESTE NÚMERO

EDITORIAL 1
INT. À LINGUAGEM MÁQUINA (Cont.) 1

Programa ZX81/Spectrum/Newbrain

- Leitura de Títulos 3
- Um Teste aos seus Reflexos 4
- Multiplicações 4
- Grande Prémio 5
- Lifter 7
- Jackpot 8
- Ficheiro Universal 10
- Novos Comandos de Execução Gráfica 13
- Header 16
- Programa de Cópia 17
- Torre de Hannoi 18

CONVERSÃO DE PROGRAMAS DO ZX81 19
NOVOS LIVROS 20
NOVOS PROGRAMAS 21

No Interior:

Cupão de Inscrição

Edição: Clube Z80

Fotocomposição: Fotomecânica Mabreu/Porto

Impressão: Ramos dos Santos & C.ª, Lda./Porto

Tiragem: 500 exemplares, Junho 1984

EDITORIAL

Autor: ROCHA BARBOSA/PORTO
PORTO

Temos insistido demasiadas vezes na necessidade e na importância do contributo dos sócios para o enriquecimento e expansão do CLUBE Z80. Mantemos como fundamental o princípio de que o Clube só pode desenvolver-se e evoluir positivamente se todos os seus elementos colaborarem nesse sentido.
Contudo, à excepção de alguns sócios — dos quais Fernando Preces merece destaque especial — a maioria permanece indiferente e inactiva, limitando-se a receber a sua publicação. Porquê?
Será que nunca criou ou adaptou um programa? Será que nunca descobriu «segredos» na sua máquina? Nunca se lembrou de escrever algumas palavras sobre os micros?
Temos a certeza que não faltarão respostas afirmativas.
Surge a questão:
Para quê guardar o produto da sua imaginação na gaveta? Porque não divulgar e dar utilidade a tudo o que é feito por si? Porque não, por exemplo, contar as suas experiências e o modo como explora o seu micro?
Há tanta coisa que poderia variar o conteúdo do CLUBE Z80!
Pense nisso:
Apostamos na sua criatividade e no seu espírito de associação. Envie sempre os seus trabalhos para publicação no CLUBE Z80.

NOTA: Ao enviar programas para publicação, não esqueça de mencionar os seus títulos, a(s) Máquina(s) em que funcionam e uma breve descrição e objectivos.

ALEXANDRE MAGALHÃES
J. MAGALHÃES
MARIA IRENE

INTRODUÇÃO À LINGUAGEM MÁQUINA — ZX81/SPECTRUM

Autor: FERNANDO PRECES

(Cont. dos números anteriores)

PARTE III — COMO FUNCIONA O Z80 (Cont.)

3.5 — 24 Registros para utilização

— **REGISTRO A** — Este é o registro mais utilizado no Z80, e também o mais importante, pois entra em todas as manipulações aritméticas ou lógicas, assim como muitas outras aplicações de menor especialização.

— **REGISTRO F** — Este registro não é utilizado para armazenar dados ou endereços como os restantes. Individualmente, os seus bits são usados como **indicadores de condições** (flags), havendo por isso quem lhe chame Registro de marcadores ou Registro de Bandeiras. Quatro desses bits (dos 8 que formam o registro) são considerados os mais importantes por desenharem funções de apoio a certas operações aritméticas ou lógicas, reflectindo as **condições dos registros intervenientes no final de cada resultado.**

Podemos considerar um flag como uma espécie de indicador (tal como uma lâmpada) que, quando está ligado (1), indica uma coisa e, quando desligado (0), indica outra.
Muitas instruções do Assembly Z80 tem como base de funcionamento o **estado** desses marcadores.
Os **4 principais FLAGS** receberam os seguintes nomes:

- **MARCADOR ZERO (zero flag)** — Este flag é afectado somente por duas condições após o resultado da última operação (aritmética ou lógica) ser acumulado no registro interveniente.

  a) Valor (1) se esse resultado for "0"
  b) Valor (0) se esse resultado for <> "0"

Tem duas mnemónicas de referência:

Z (zero) para flag = 1 (set)
NZ (nota zero) para flag = 0 (reset)

Todas as instruções de comparação e as suas condicionadas baseiam-se no **estado** (set ou reset) deste flag.

- **MARCADOR DE SINAL (sign flag)** — Este flag é efectado por todas as operações efectuadas em **2.º complemento aritmético.** Como devem estar lembrados, neste tipo de sistema aritmético o bit 7 (o último num registro de 8 bits) é reservado para indicar a polaridade (negativa ou positiva) dum número binário. O sign flag reflecte esta condição para o resultado da última operação acumulada no registro interveniente.

  a) Valor (1) se o resultado for negativo (−)
  b) Valor (1) se o resultado for positivo (+)

Tem duas mnemónicas de referência:

M (minus) para flag = 1 (set)
P (positive) para flag = 0 (reset)

- **MARCADOR DE TRANSBORDO (carry flag)** — Quando um registro atinge um máximo de contagem (para um registro de 8 bits, o binário 1111 1111), não pode sem auxílio exterior, continuar a acumular resultados porque a partir daí, basta que lhe seja adicionado o menor número positivo (binário 0000 0001) para que nele se dê um **estouro de acumulação,** voltando ao ponto de partida (binário 0000 0000), perdendo portanto toda a informação anterior.
Se ficar confuso com o que acaba de ler, lembre-se como se comporta um conta-rotações mecânico ou electrónico, que acumula até ao máximo dos seus digitos e depois volta a zero, ou experimente este pequeno programa em BASIC e poderá compreender melhor o que significa um estouro de acumulação:

```
10 FOR n=200 TO 300
20 POKE 16507,n          (Para o ZX81)
20 POKE 23300,n          (Para o Spectrum)
```

```
30 PRINT AT 12,0; "VALOR ACUMULADO —"; AT 12,20;
   PEEK 16507        (ou 23300)
40 FOR m=0 TO 20
50 NEXT m
60 PRINT AT 12,20;"    "    (3 espaços)
70 NEXT n
```

Pode verificar, quando rolar o programa, que a célula de memória carregada com o valor de n, acumula resultados até atingir o número 255, voltando a zero na solicitação seguinte, partindo para uma nova contagem e perdendo a anterior.
A missão fundamental deste flag é tomar o valor (1) sempre que aconteça um estouro de acumulação após o resultado de uma dada operação ser adicionada a um registro.

Tem 2 mnemónicas de referência:

C (carry set) para flag = 1
N (no carry) para flag = 0

Este flag ainda é utilizado em certas instruções de subtracção, de rotação e outras.

- **MARCADOR DE PARIDADE/DEFEITO DE ACUMULAÇÃO (Parity/over flow flag)** — Este é um flag que serve para testar duas ocorrências em cada um dos seus estados.

É usado para testar um **defeito de acumulação,** quando um dado é armazenado num registro em 2.º complemento aritmético, vigiando o bit de sinal que pode ser adulterado se os restantes bits tiverem um estouro de acumulação. Este flag toma o valor (1) sempre que o bit de sinal é afectado.
É usado para testar a **paridade** no resultado de certas operações lógicas, contando o número de **uns** existentes no registro interveniente.
Flag = 1 (set) — Quando em 2.º complemento aritmético o bit de sinal é afectado ou em operações lógicas o resultado tiver um número **par** de **uns.**
Flag = 0 (reset) — Quando exista um número **ímpar** de **uns.**
Tem duas mnemónicas de referência:

PE (party even) para flag = 1
PO (party odd) para flag = 0

Os restantes flags, dois dos quais têm nomes atribuídos **(N — add/subtract flag)** e **(H — half carry flag),** são utilizados pela unidade de control e não podem ser manipulados por qualquer tipo de instrução proveniente do assembly, pelo que não voltarão a ser referenciados por este texto.

A posição de cada flag, dentro do registo F é a seguinte:

7	6	5	4	3	2	1	0
S	Z	X	H	X	P/V	N	C

posição dos bits no registro F

bit 0 — C — carry flag
» 1 — N — add/subtract flag
» 2 — P/V — Parity/over flow flag
» 3 — X (sem nome)
» 4 — H — half-carry flag
» 5 — X — (sem nome)
» 6 — Z — zero flag
» 7 — S — sign flag

— **REGISTRO CONTADOR (PAR BC)** — o registro B, e por extensão o C (com o qual forma par), trabalha com muitas instruções do assembly Z80. No entanto a sua função mais especializada é como registro **contador.**
Em linguagem máquina, a contagem de impulsos para um ciclo semelhante ao FOR-NEXT do Basic é efectuada pelo registro BC.
Algumas instruções permitem o trabalho individual, quer da célula B, quer da C.
Este registro tem ainda a seguinte particularidade:

a) **Transporta do Basic** o endereço de arranque da rotina máquina.
b) **Transporta para o Basic** qualquer dado ou endereço, quando a rotina máquina terminar com uma instrução RET (muito semelhante a RETURN do Basic).

— **REGISTROS DE ENDEREÇOS DE e HL** — Além do trabalho que as células D, E, H e L podem executar individualmente, como registros de 8 bits, os pares DE e HL são muito utilizados em linguagem máquina, na retenção de endereços da memória para funções muito diferenciadas.
O registro HL é porém o mais utilizado, porque pode executar um certo número de operações aritméticas, como interveniente principal, acumulando o resultado.

— **REGISTROS DE INDEXAÇÃO IX e IY** — Estes dois registros são células inteiras de 16 bits, não sendo possível, em contraste com os registros BC, DE e HL, a sua separação para trabalho individual.
São muito importantes porque, após a retenção de um determinado endereço, são capazes de percorrer (mediante a variável introduzida) 255 endereços para a frente, memorizar o conteúdo de qualquer deles e continuarem fixados sobre o endereço de origem. Esta característica dá-lhes a possibilidade de trabalharem sobre uma tabela de dados, de caracteres ou qualquer outro bloco de informação especializada, a uma enorme velocidade. Um inconveniente se pode apontar e que exige do programador uma certa atenção: necessitam de retornar à ROM (algumas rotinas monitoras assim o determinam) com os endereços que tinham memorizados quando de lá sairam.

— **REGISTRO APONTADOR DA PILHA (stack pointer)** — A pilha é uma área junto do topo da RAM que serve para armazenagem temporária do conteúdo dos registros pares, incluindo o par especial AF. O seu primeiro endereço fica imediatamente abaixo do apontado pela variável RAMTOP e cresce, movimentando-se para trás, descendo os endereços da RAM.
Cada vez que o conteúdo dum registro é introduzido no Stack, o apontador SP é decrementado 2 vezes: a primeira para a entrada do **High byte** (1.ª letra dum registro par) e a segunda para a entrada do **Low byte** (2.ª letra do registro).
É incrementado 2 vezes na acção inversa, ou seja quando um registro vai buscar essa informação ao Stack.

— **REGISTROS ALTERNATIVOS AF′**, BC′, DE′ e HL′ — existe no Z80 um duplicado dos quatro principais registros pares.
As vantagens da sua utilização podem ser espantosas, porque as instruções de **troca** de conteúdos entre os registros principais e os registros alternativos são executados muito mais rapidamente do que as instruções de armazenagem de conteúdos no Stack.
Porém alguns inconvenientes, que serão estudados na devida altura, forçam o programador a evitar por vezes a sua utilização.

— **REGISTRO I** — A execução **normal** das instruções que formam um programa é feita no Z80 de uma forma sequencial (em Basic, partindo da primeira instrução da linha n; em código máquina, partindo da instrução contida no endereço n). Essa sequência pode ser desviada a determinado passo, para uma outra (quer em Basic, quer em cm.) por qualquer salto endereçado pelo programa, ou ainda por um **sinal de interrupção** que obriga o processador a percorrer uma outra combinação sequencial, normalmente monitorada na ROM, durante a qual fica aparentemente imobilizado, aguardando um sinal de desinibição.
Na generalidade dos microcomputadores que possuem um Z80, quando ocorre uma interrupção, o registro I (de 8 bits) será fixado sobre a parte alta do endereço (High Byte) da rotina de tratamento de interrupções, enquanto que o elemento causador da interrupção é memorizado com os bits da parte baixa desse endereço (Low Byte).
A reunião desses 16 bits compõem um endereço vector, que aponta a correspondente subrotina de interrupção.
As ROMS do ZX81 e do Spectrum possuem 2 tipos de subrotinas de interrupção (endereços 0038 e 0066 Hex.), mas o Spectrum só utiliza pelos processos usuais a rotina (endereço 0038), sendo condicionada a utilização da rotina (endereço 0066).
Estes dois computadores dão uma utilização especial ao registro I, como apontador da tabela de caracteres para a informação de vídeo. É por essa razão, que se torna possível (é necessário copiar para a RAM, algumas rotinas monitoras da ROM) pôr o ZX81 a imprimir no écran caracteres definidos pelo utilizador.

— **REGISTRO R** — Nas versões anteriores ao ZX81, este registro de 8 bits, era usado para **restaurar** zonas da memória RAM aonde os sinais binários estivessem com tendência à indefinição (nível de voltagem inferior à necessária para se diferenciar o **estado 1** do **estado 0**).
Com o avanço da tecnologia, as memórias vivas deixaram de precisar de tais cuidados, tendo o ciclo máquina sido encurtado em duração, o que se traduz num aumento significativo da velocidade de processamento.
Nestes novos sistemas, nos quais estão incluídos o ZX81 e o SPECTRUM, o registro R é aproveitado para contar o número de caracteres que deverão formar a próxima linha horizontal de vídeo.

3.6 — Unidade de aritmética lógica (ALU)

Este é o último dos blocos funcionais do Z80. Tal como o seu nome indica, esta unidade foi concebida para tratar de operações aritméticas ou lógicas, mas com um raio de acção muito limitado.
Pode efectuar operações de adição e subtracção, mas nem divide nem multiplica, visto que estas operações implicam uma lógica muito mais complexa em binário aritmético. Trata também as operações lógicas AND, OR e XOR que sejam introduzidas em comando directo, através de um grupo especializado de instruções Assembler, bem como operações de **bits.**

(Continua no próximo número)

LEITURA DE TÍTULOS

ZX81

Autor: ROCHA BARBOSA/PORTO
PORTO

Este programa faz a leitura dos títulos dos programas gravados em cassete, dando uma lista dos mesmos. É, portanto, útil para os esquecidos.

Introduza o programa, faça RUN, aguarde que apareça a indicação 9/60, por intermédio de EDIT apague as linhas de 5 a 60 (inclusive) e faça RUN 100.

O ZX81 aguardará que introduza no gravador a cassete que pretende «ler» e, após o começo da reprodução, listará o(s) título(s) do(s) programa(s).

NOTA: Se pretender mudar de cassete, pode fazê-lo sem interromper o programa e os títulos da nova cassete irão aparecendo na lista que o programa lhe faz.

```
   1 REM 00000000010 ... (96 car
acteres)
   5 FOR I=16515 TO 16609
  10 POKE I,PEEK (I-15680)
  15 NEXT I
  20 POKE 16514,55
  25 POKE 16520,140
  30 POKE 16521,64
  35 POKE 16536,35
  40 LET A$="1114FF2A10400106000
9CD8C407123CB79C01D20F5C9C3A203"
  45 FOR I=1 TO 25
  50 POKE 16549+I,16*CODE A$(2*I
-1)+CODE A$(2*I)-476
  55 NEXT I
  60 STOP
 100 CLEAR
 105 DIM B$(20)
 110 FAST
 115 RANDOMIZE USR 16514
 120 PRINT B$
 125 PAUSE 200
 130 POKE 16437,255
 135 RUN 100
```

PROGRAMAS MAIS VENDIDOS

Sempre que possível, tentaremos, a partir de agora, fornecer aos leitores uma lista dos programas mais solicitados ao CLUBE Z80, no mês anterior ao da publicação da revista. Assim, no mês de Maio, os TOPS foram:

1 — FIGHTER PILOT
2 — CHEQUERED FLAG
3 — JET SET WILLY
4 — NIGHT GUNNER
5 — HUNTER KILLER
6 — SIMULADOR DE VOO (PSION)
7 — SPACE SHUTTLE
8 — PINBALL
9 — ATIC ATACK
10 — PARATROOPERS

UM TESTE AOS SEUS REFLEXOS

Autor: FERNANDO PRECES

```
  1 REM "F2J1"
  2 PRINT ,," UM TESTE AOS SEUS REFLEXOS"
  3 GO SUB 500
  4 FOR N=1 TO 500
  5 NEXT N
  8 CLS
 10 LET N=0
 20 FOR M=1 TO 20
 30 FOR B=0 TO RND*100+50
 40 NEXT B
 50 LET B$=CHR$ (28+35*RND)
 60 PRINT AT 12,15;B$
 70 LET C$=""
 80 FOR B=0 TO 80
 90 NEXT B
100 LET C$=INKEY$
110 IF C$=B$ THEN GO SUB 200
120 CLS
130 NEXT M
140 PRINT "PONTUACAO=";N
150 PRINT ,,"PRIMA (N/L)"
160 INPUT I$
163 PAUSE 40
165 CLS
166 PAUSE 20
170 GO TO 10
200 LET N=N+1
210 PRINT AT 14,13;"CERTO"
220 FOR B=0 TO 20
230 NEXT B
240 RETURN
500 PRINT ,,,,"AO ACASO VAI APARECER ECRAN UM"
505 PRINT ,,"SIMBOLO, LETRA OU NUMERO, QUE VO"
510 PRINT ,,"CE TERA DE  REPETIR CARREGANDO A"
515 PRINT ,,"TEMPO, NA TECLA RESPECTIVA."
520 PRINT ,,,,,," ESPERE A PONTUACAO  E TENTE MEL"
525 PRINT ,,"HORA-LA.   BOA SORTE."
530 RETURN
```

MULTIPLICAÇÃO

ZX81

```
  2 REM "F2J2"
  3 CLS
  5 PRINT ,,"JOGO DIDATICO"
 10 LET Z=0
 20 GO SUB 460
 30 PRINT "QUESTIONARIO DE MULTIPLICACAO "
 40 GO SUB 460
 50 PRINT "GRAU DE DIFICULDADE (1 A 10)  QUAL ESCOLHE ?"
 60 INPUT A
 70 IF A<1 OR A>10 THEN GO TO 60
100 GO SUB 460
110 PRINT "QUANTAS PREGUNTAS ?"
120 INPUT B
130 IF B<1 THEN GO TO 120
140 CLS
150 FOR G=1 TO 8
160 LET C=A*INT (RND*10)+1
170 LET D=A*INT (RND*10)+1
180 LET E=C*D
190 GO SUB 460
200 PRINT "PREGUNTA NUMERO ";G
210 GO SUB 460
220 PRINT "QUANTO E ";C;" VEZES ";D;"?"
230 INPUT F
240 PRINT ,,"= ";F
250 GO SUB 460
260 IF F=E THEN GO TO 420
270 PRINT "INCORRECTO, A RESPOSTA CERTA E: << ";E;" >>"
280 GO SUB 460
290 PRINT TAB 4;" << PONTUACAO >> ";Z
295 PRINT
300 PRINT TAB 5;"RESPOSTAS DADAS ";G
301 LET B=B-1
302 IF B=0 THEN GO TO 390
305 PRINT
310 PRINT "PRIMA (N/L) PARA CONTINUAR";
330 INPUT A$
340 CLS
350 NEXT G
360 FOR K=1 TO 3
370 GO SUB 460
380 NEXT K
390 PRINT "ACABOU O QUESTIONARIO. O SEU RECORDE E DE:"
400 PRINT TAB 8;INT (Z*100/G);" / CENTO"
410 GO TO 500
420 LET Z=Z+1
430 GO SUB 460
440 PRINT "OK...RESPOSTA CORRECTA."
450 GO TO 280
460 FOR S=1 TO 2
470 PRINT
480 NEXT S
490 RETURN
500 FOR N=1 TO 300
510 NEXT N
520 CLS
530 PRINT AT 12,0;"QUER CONTINUAR? (DIGA S OU N)"
540 INPUT L$
550 IF L$="S" THEN RUN
560 PRINT ,,"OBRIGADO ... BYE.. BYE"
900 SAVE "F2J2" LINHA 1
```

GRANDE PRÉMIO

SPECTRUM 16 K

In. HOME COMPUTING
Trad. e Adapt.: ANTÓNIO AMARAL/Porto

```
          GRANDE PREMIO
 Bemvindo a pista.Voce,o piloto
do carro de corrida,deve
percorrer a pista evitando os
obstaculos como      ,

 Tambem tera que guiar de noite,
evitar os trabalhos na estrada
e ter cuidado nos sitios
estreitos.
 A intervalos tambem tera de
encher o deposito do carro
A pontuacao esta a esquerda e
o numero de vidas a direita
 CONTROLES:  5 ESQUERDA
             8 DIREITA
 AS TECLAS 1,2,3 E 4 metem as
mudancas de caixa respectivas
```

```
  1 FOR f=USR "a" TO USR "u"+7:
 READ a: POKE f,a: NEXT f
  2 CLS : PRINT AT 18,15; INK 2
;"  ";AT 19,15; INK 2;"  ";AT 20
,15; INK 2;"  "
  3 LET a$="          GRANDE PR
EMIO          Bemvindo a pista.V
oce,o piloto do carro de corrida
,deve         percorrer a pista e
vitando os    obstaculos como
  ,
              Tambem tera que gu
iar de noite,evitar os trabalhos
 na estrada   e ter cuidado nos s
itios         estreitos.
              A intervalos tambe
m tera de     encher o deposito d
o carro       A pontuacao esta a
 esquerda e  o numero de vidas a
 direita      CONTROLES: 5 ESQUE
RDA                     8 DIREI
TA            AS TECLAS 1,2,3 E
4 metem as    mudancas de caixa r
espectivas    "
  4 PRINT AT 0,0;: FOR f=1 TO L
EN a$: PRINT a$(f);: BEEP .01,RN
D*60: BEEP .01,RND*60+3: NEXT f
  5 LET h1=40: LET h2=80: LET h
3=140
  6 LET l=19
  7 FOR f=-30 TO 50: BEEP .01,f
: BEEP .01,f+3: BEEP .01,f+6: NE
XT f
  8 FOR f=0 TO 21: PAPER 0: INK
 7: BORDER 0: RANDOMIZE USR 3280
: NEXT f
  9 LET a=0: LET d=0: LET j=0:
LET attr=60: PAPER 0: INK 7: BOR
DER 0
 25 LET li=3: LET x=16: LET y=1
0: LET s=0
 30 INPUT "NIVEL DE DIFICULDADE
 (6-15)     15 = FACIL   6 = DI
FICIL      ";u
 40 IF u<6 OR u>15 THEN GO TO 3
0
 50 PRINT AT y,x; INK 2;"  ":
 PRINT AT y+1,x; INK 2;"  ": P
RINT AT y+2,x; INK 2;"  "
 60 PRINT AT y-1,x;"  "
 70 PRINT AT 0,12;s;AT 0,17;"↑↑
↑↑"( TO li)
 80 LET x=x+(INKEY$="8")-(INKEY
$="5")
 81 BEEP .01,1/4
 85 IF ATTR (13,x)=attr THEN GO
 TO 8000
 90 IF x=11 OR x=22 THEN GO TO
8000
 91 IF a=2 OR a=4 OR a=6 OR a=8
 THEN PRINT AT 20,0;"      
"
 92 IF a=2 OR a=4 OR a=6 OR a=8
 THEN PRINT AT 20,23;"     
"
100 LET v=USR 3280
110 PLOT 100,0: DRAW 0,8: PLOT
180,0: DRAW 0,8: FLASH 0
115 IF li<=0 THEN GO TO 4500
116 LET a=a+1
117 IF INKEY$="1" THEN LET l=18
: BEEP .01,3
118 IF INKEY$="2" THEN LET l=17
: BEEP .01,3
119 IF INKEY$="3" THEN LET l=16
: BEEP .01,3
120 IF INKEY$="4" THEN LET l=15
: BEEP .01,3
125 IF a=u THEN GO TO 1000
135 IF s=20 OR s=200 THEN GO TO
3000
136 IF s=140 OR s=50 THEN GO TO
2000
137 IF s=70 OR s=10 THEN LET j=
0: GO TO 2050
138 IF s=90 THEN GO TO 4000
140 IF s=160 THEN GO TO 2200
150 IF s=120 THEN GO TO 3500
160 IF s=1 THEN GO TO 4000
170 LET s=s+.5
180 GO TO 50
1000 PRINT AT l,x; INK 4;"  ";A
T l+1,x; INK 4;"  ";AT l+2,x; I
NK 4;"  "
1001 PRINT AT 20,0;"      "
1002 PRINT AT 20,23;"      "
1010 LET a=0
```

```
1020 GO TO 50
2000 PRINT AT 19,16; INK 4;"[illegible]"
2001 PRINT AT 20,16; INK 4;"[illegible]"
2002 LET s=s+5
2010 GO TO 50
2050 PRINT AT 19,17; INK 4;"[illegible]
[illegible]": PRINT AT 20,17; INK 4;"[illegible]
"
2060 LET s=s+10: GO TO 50
2200 INK 2: PLOT 120,0: DRAW 0,1
75: PLOT 160,0: DRAW 0,175: INK
0
2201 IF x<13 OR x>19 THEN GO TO
8000
2202 LET d=d+1: IF d=u*3 THEN LE
T s=s+20
2300 GO TO 50
3000 INK 4: PLOT 180,10: DRAW -6
0,10: DRAW 5,20: DRAW 55,0: INK
0
3010 PRINT AT 17,16; FLASH 1; BR
IGHT 1;"OBSTACULO"
3020 LET s=s+5: GO TO 50
3500 PLOT 140,0: DRAW 0,20: PLOT
 160,0: DRAW 0,20
3510 IF j=1 THEN PRINT AT 18,18;
"[illegible]";AT 19,18;"[illegible]";AT 20,18;"[illegible]";AT
 21,18;"[illegible]"
3515 IF j=5 THEN GO TO 3520
3517 GO TO 3530
3520 IF x<=15 OR x>=18 THEN LET
li=0: GO TO 3560
3530 LET j=j+1: IF j=20 THEN LET
 s=s+10: LET j=0: LET li=li+1
3550 LET a=2
3560 GO TO 50
4000 IF j<30 THEN LET attr=4
4001 IF j<30 THEN PAPER 0: INK 7
4002 LET j=j+.5
4009 IF j=30 THEN LET attr=60
4010 IF j=30 THEN LET s=s+10
4011 IF j=30 THEN PAPER 7: INK 0
: BORDER 7
4020 GO TO 50
4500 IF s<h1 THEN GO TO 5040
4501 PAPER 7: BORDER 7: INK 0
4510 CLS : LET a$="Muito bem,ati
ngiu um novo maximoEscreva o seu
 nome."
4511 PRINT AT 10,0;: FOR i=1 TO
LEN a$: BEEP .01,RND*30+3: PRINT
 INK RND*6; FLASH 1;a$(i);: NEXT
 i
4520 POKE 23658,8: INPUT n$: POK
E 23658,0
5002 IF s>h1 AND s>h2 THEN LET h
1=s
5003 IF s>h2 AND s<h3 THEN LET h
2=s
5004 IF s>h3 THEN LET h3=s
5010 CLS : PRINT AT 0,10;"QUADRO
 DE PONTUACAO": PRINT AT 5,10;"1
-",h3;AT 10,10;"2-",h2;AT 15,10;
"3-",h1
5020 IF h2=s THEN PRINT AT 10,23
;n$
5021 IF h1=s THEN PRINT AT 15,23
;n$
5022 IF h3=s THEN PRINT AT 5,23;
n$
5030 FOR f=0 TO 600: NEXT f
5040 CLS : PRINT AT 10,10; INK R
ND*6; FLASH 1;"OUTRO JOGO ?": BE
EP .1,15
5050 IF INKEY$="" THEN GO TO 504
0
5060 IF INKEY$="s" OR INKEY$="S"
 THEN GO TO 6
5070 STOP
8000 FOR f=0 TO 300: OUT 16,38:
OUT 38,16: NEXT f
8010 FOR f=0 TO 30: BEEP .01,f:
BEEP .01,f+6: BEEP .01,f*2: NEXT
 f
8011 FOR f=30 TO 0 STEP -1: BEEP
 .01,f+6: BEEP .01,-f: BEEP .01,
-f: NEXT f
8015 LET x=15: CLS
8020 LET a=0: LET li=li-1
8030 GO TO 50
9000 DATA 255,240,31,7,227,239,2
35,248
9001 DATA 255,15,248,224,199,247
,215,31
9002 DATA 233,233,232,12,15,12,7
,7
9003 DATA 151,151,23,48,240,48,2
24,224
9004 DATA 3,99,97,111,120,111,96
,96
9005 DATA 192,198,134,246,30,246
,6,6
9006 DATA 8,28,28,62,62,127,8,8
9007 DATA 0,7,15,31,31,17,53,117
9008 DATA 0,0,254,255,255,95,95,
95
9009 DATA 117,113,127,127,63,31,
0,0
9010 DATA 94,66,254,254,255,255,
248,0
9011 DATA 0,0,0,0,31,35,91,107
9012 DATA 0,0,0,7,255,255,0,255
9013 DATA 0,0,15,255,255,240,15,
255
9014 DATA 0,3,255,255,252,3,255,
255
9015 DATA 0,248,252,254,7,255,25
5,3
9016 DATA 165,213,201,171,83,71,
63,0
9017 DATA 255,0,255,255,0,255,24
0,0
9018 DATA 240,15,255,0,255,240,0
,0
9019 DATA 3,255,192,63,248,0,0,0
9020 DATA 255,254,6,254,0,0,0,0
```

```
GRAFICOS:

Linha 2, 50, 1000
- G + teclas A,B/C,D/E,F

Linha 91   - G + tecla G

Linha 2000- G + teclas H,I

Linha 2001- G + teclas J,K

Linha 2050- G + teclas L,M,N,O,P
                       Q,R,S,T,U

Linha 3510 - G + teclas F/U/E
```

Quando quiser verificar se o SPECTRUM é 16 ou 48 K:

PRINT PEEK 23733 (Enter)

Se 255 — 48 K
Se 127 — 16 K

LIFTER

SPECTRUM 16 e 48 K

In. YOUR COMPUTER, Março 1983
Trad. e Adapt.: J. MAGALHÃES

DESCRIÇÃO

Este jogo consta, como mostra a fig. 4, de uma construção com vários níveis, no último dos quais se encontra JANE que foi capturada por um gorila.

Para salvar JANE, terá de percorrer todos os níveis saltando as bolas de fogo e usando o elevador para ir de um nível para outro. Quando passa por uma ponte, esta desaparece deixando um buraco que tem de evitar saltando, ou perderá uma vida.

As teclas M e N permitem-lhe deslocar para a direita e para a esquerda, e a tecla CAPS SHIFT permite-lhe saltar sobre as bolas de fogo. Só terminará uma fase depois de destruir as pontes de todos os níveis.

INTRODUÇÃO DO PROGRAMA

1. Passe a listagem do programa em BASIC e grave da seguinte forma:
   SAVE «LIFTER» LINE 9999
2. Passe a rotina da fig. 1; RUN; dê entrada dos 128 bytes da fig. 2; grave:
   SAVE «USER DEF» CODE USR «a», 128

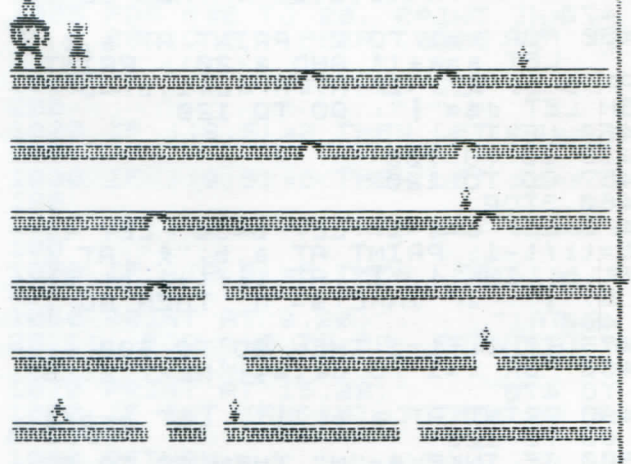
NOTA: O programa em código máquina deve ser gravado logo a seguir ao programa em BASIC.
Para passar o jogo, apenas deve fazer: LOAD »» (na linha 9999 encontra--se a chamada ao cód. máq. LOAD »» CODE) e esperar a passagem dos 2 programas (BASIC+Cód. Máq.).

```
  10 FOR n=USR "a" TO (USR "p"+7
)
  20 INPUT c: POKE n,c
  30 PRINT n,c
  40 NEXT n
```

```
24   24   48   124  176  56   100
70   24   24   12   62   13   28
38   98   153  189  189  153  126
60   60   60   60   126  126  255
36   36   36   102  255  255  231
195  129  129  129  0    3    7
5    6    4    6    31   127  128
192  64   192  64   192  240  252
255  251  209  208  216  216  121
124  254  191  23   19   51   51
54   118  31   31   60   56   56
24   56   120  240  240  120  56
56   48   56   60   255  0    221
85   85   85   119  0    255  126
24   24   24   24   24   24   24
24   24   24   24   24   24   24
8    24   36   40   24   60   60
24   60   66   129  129  129  129
66   60
```

```
Os graficos estao definidos
pelas teclas:

B -        H -        N -
C -        I -        O -
D -        J -        P - O
E -        K -
F -        L -
```

SCORE-0050 PARTE-1 VIDAS-

```
  10 CLS : PRINT AT 8,12;"N....E
squerda";AT 10,12;"M....Direita"
;AT 14,3;"CAPS SHIFT....Saltar":
 PAUSE 500
 100 LET m=1: LET lift=20: LET d
f1=1: LET df2=0: LET df3=0: LET
df4=1: LET delay=20: LET sheet=0
: LET sc=0: LET lives=5: BORDER
0: PAPER 0: INK 7: CLS : GO SUB
9000
 110 LET a=18: LET b=2: PRINT AT
 a,b;""
 120 IF INKEY$="M" OR INKEY$="N"
 THEN GO SUB 500
 140 IF INKEY$="m" AND b<31 THEN
 LET b=b+1: PRINT AT a,b-1;" "
 150 IF INKEY$="n" AND b>1 THEN
LET b=b-1: PRINT AT a,b;" "
 153 IF (a=3 AND b=f1+1) OR (a=9
 AND b=f2+1) OR (a=15 AND b=f3+1
) OR (a=18 AND b=f4+1) THEN GO T
O 600
 155 IF ATTR (a+1,b)<>3 THEN GO
TO 400
 160 LET f1=f1+(df1=1)-(df1=0):
IF df1=0 AND f1<5 THEN LET df1=1
 164 IF df1=1 AND f1>26 THEN LET
 df1=0
 166 PRINT AT 3,f1; INK 2;f$
 170 LET f2=f2+(df2=1)-(df2=0):
IF df2=0 AND f2<4 THEN LET df2=1
 174 IF df2=1 AND f2>23 THEN LET
 df2=0
 176 PRINT AT 9,f2; INK 2;f$
 180 LET f3=f3+(df3=1)-(df3=0):
IF df3=0 AND f3<4 THEN LET df3=1
 184 IF df3=1 AND f3>25 THEN LET
 df3=0
 186 PRINT AT 15,f3; INK 2;f$
 190 LET f4=f4+(df4=1)-(df4=0):
IF df4=0 AND f4<4 THEN LET df4=1
 194 IF df4=1 AND f4>22 THEN LET
 df4=0
 196 PRINT AT 18,f4; INK 2;f$
 200 LET lift=lift-1: IF lift<1
THEN PRINT AT lift+1,31; INK 5;"
": LET lift=20
 210 PRINT AT lift,31; INK 4;""
;AT lift+1,31; INK 5;""
 215 IF ATTR (a,b-1)=6 AND sc=(1
20*sheet) THEN PRINT AT a,b;" ":
 GO SUB 8000: GO SUB 9000: GO TO
```

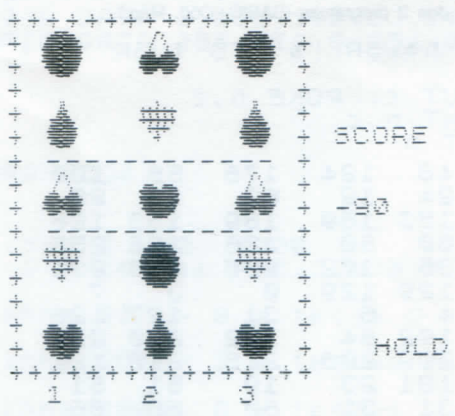
```
 110
 220 GO TO 120
 400 IF ATTR (a+1,b)=5 THEN PRIN
T AT a+1,b;" ": LET sc=sc+(10 AN
D b<30): PRINT AT 0,6;"0000"( TO
 4-LEN STR$ sc);sc: GO TO 120
 410 IF ATTR (a+1,b)=7 THEN GO T
O 450
 420 IF ATTR (a+1,b)=4 THEN GO T
O 470
 450 FOR x=1 TO 2: PRINT AT a,b;
" ": LET a=a+(1 AND a<20): PRINT
 AT a,b;"Ж": IF ATTR (a+1,b)=3 T
HEN LET d$="▌": GO TO 120
 452 NEXT x
 455 GO TO 700
 457 GO TO 120
 460 STOP
 470 LET a=a-1: LET b=31: LET li
ft=lift-1: PRINT AT a,b;"Ж";AT l
ift,b; INK 4;"T";AT lift+1,b; IN
K 5;"▌": IF INKEY$="n" THEN GO T
O 490
 475 IF lift=1 THEN GO TO 600
 480 FOR x=1 TO delay: NEXT x: G
O TO 470
 490 PRINT AT a,b;"▌": LET b=b-1
: GO TO 120
 500 IF INKEY$="M" THEN GO TO 52
0
 510 IF INKEY$="N" THEN GO TO 53
0
 515 RETURN
 520 PRINT AT a-1,b;"Ж";AT a,b;"
": LET b=b+1: PRINT AT a-1,b;"Ж
";AT a-1,b-1;" ": LET b=b+1: PRI
NT AT a-1,b;"Ж";AT a-1,b-1;" ";A
T a,b;"Ж";AT a-1,b;" ": RETURN
 530 PRINT AT a-1,b;"Ж";AT a,b;"
": LET b=b-1: PRINT AT a-1,b;"Ж
";AT a-1,b+1;" ": LET b=b-1: PRI
NT AT a-1,b;"Ж";AT a-1,b+1;" ";A
T a,b;"Ж";AT a-1,b;" ": RETURN
 600 FOR x=90 TO 65 STEP -1: PRI
NT AT a,b; OVER 1;CHR$ x: BEEP .
01,x-50: NEXT x: PRINT AT a,b;"
": GO TO 800
 700 FOR x=a TO 21: PRINT AT x,b
;"Ж";AT x-1,b;" ": BEEP .01,x: N
EXT x: FOR y=90 TO 65 STEP -1: P
RINT AT x-1,b; OVER 1;CHR$ y: BE
EP .01,y-50: NEXT y: PRINT AT x-
1,b;" "
 800 LET lives=lives-1: PRINT AT
 0,25;"ЖЖЖЖЖ"( TO lives);" "
 805 BEEP .25,5: BEEP .25,2: BEE
P .1,7: BEEP .3,5: BEEP .3,2
 810 IF lives>0 THEN GO TO 110
 820 PRINT AT 19,10; INK 6; FLAS
H 1;"FIM DO JOGO";AT 21,3; FLASH
 0; INK 4;"Tecla ""R"" novo jogo
"
 830 IF INKEY$<>"r" THEN GO TO 8
30
 840 RUN
8000 RESTORE 8010: FOR x=1 TO 22
: READ b,a: BEEP a/30,b: NEXT x
8010 DATA 0,3,2,3,4,3,5,5,0,6,5,
3,4,3,5,3,7,5,2,6,5,3,4,3,5,3,9,
3.5,7,1.5,7,4,5,3,5,3,4,3,2,4,4,
4,5,5
8020 RETURN
9000 FOR x=3 TO 20 STEP 3: PRINT
 AT x,0;"
                   ": NEXT x
9005 FOR x=4 TO 20 STEP 3: PRINT
 AT x,0; INK 3;"████████████████
████████████████": NEXT x: FOR x=
3 TO 21: PRINT AT x,31; INK 5;"▌
": NEXT x: PRINT AT 20,31; INK 4
;"T"
9007 FOR x=4 TO 20 STEP 3
9010 LET r=INT (RND*20)+5: LET t
=INT (RND*20)+5: IF r=t OR r+2=t
 OR r-1=t THEN GO TO 9010
9015 PRINT AT x,r; INK 5;"Π";AT
x,t; INK 5;"Π": NEXT x
9020 PRINT AT 18,2;"Ж"
9030 GO SUB 9100: LET f1=6: LET
f2=15: LET f3=6: LET f4=9: INK 2
: PRINT AT 3,f1;f$;AT 9,f2;f$;AT
 15,f3;f$;AT 18,f4;f$: INK 7
9040 PRINT AT 1,0; INK 6;"Д";AT
 2,0;"ΠΠ Ψ";AT 3,0;"Π Т"
9060 LET sheet=sheet+1: LET dela
=delay-1
9070 PRINT AT 0,0;"SCORE-";"0000
"( TO 4-LEN STR$ sc);sc;" PARTE-
";sheet;" VIDAS-";"ЖЖЖЖЖ"( TO li
ves);" "
9080 RETURN
9100 IF m=1 THEN LET f$=" ¥ ": L
ET m=m+1: RETURN
9110 IF m=2 THEN LET f$=" O ": L
ET m=m+1: RETURN
9120 IF m=3 THEN LET m=1: GO TO
9100
9999 LOAD ""CODE : RUN
```

JACKPOT

SPECTRUM 16 e 48 K

In. YOUR COMPUTER, Março 1983
Trad. e Adapt.: J. MAGALHÃES

É um programa de uma máquina de frutas. Estas rodam, accionando a tecla 0 e o **Hold** pelas teclas 1, 2 e 3 quando lhe é permitido (fixa qualquer rolo).

Para um maior grau de dificuldade, altere os valores das linhas 320 e 1080 (3 e 5).

```
TECLA 0 COMECAR
TECLA 9 CANCEL HOLD

   8 PRINT AT 16,3;"+++++++++++
++++"
  10 PRINT AT 1,3;"++++++++++++
+++"
  12 FOR n=2 TO 15: PRINT AT n,3
;"+": PRINT AT n,18;"+": NEXT n
  14 PRINT AT 7,5;"------------"
  16 PRINT AT 10,5;"------------
"
  18 PRINT AT 6,20;"SCORE": LET
sc=0
  20 PRINT AT 19,3;"TECLA 0 COME
CAR": PRINT AT 20,3;"TECLA 9 CAN
CEL HOLD"
  28 FOR i=97 TO 112: PRINT AT 0
,ABS ((104-i)*2)+3;"©": BEEP .01
25,0: FOR n=0 TO 7
```

```
  30 PRINT AT 0,ABS ((104-i)*2)+
3;" ": READ x: POKE USR CHR$ i+n
,x
  32 NEXT n: NEXT i
  34 DIM j(18,18)
  36 DIM k(15): DIM l(15): LET l
(5)=1: LET l(10)=2: LET l(15)=3
  38 LET p=5: LET q=15: LET r=5
  40 BEEP .0125,40: BEEP .0125,4
0: BEEP .0125,40: PRINT AT 15,22
;"    ";AT 17,5; INK 1;"1    2
  3": LET sc=sc-10: PRINT AT 9,2
0;"      ";AT 9,20;sc
  50 FOR x=p TO q STEP r
  55 LET k(x)=INT (RND*5)+1
  60 IF k(x)=l(x) THEN GO TO 55
  65 LET l(x)=k(x): LET y=3*k(x)
  70 INK 4: LET j(y,x)=1: BEEP .
01,-10
  75 PRINT AT y,x;"  "
  80 PRINT AT y-1,x;"  "
  85 IF y=15 THEN LET y=0
  90 INK 2: LET j(y+3,x)=2: BEEP
 .01,-10
  95 PRINT AT y+3,x;"  "
 100 PRINT AT y+2,x;"  "
 105 IF y=12 THEN LET y=-3
 110 INK 6: LET j(y+6,x)=3: BEEP
 .01,-10
 115 PRINT AT y+6,x;"  "
 120 PRINT AT y+5,x;"  "
 125 IF y=9 THEN LET y=-6
 130 INK 2: LET j(y+9,x)=4: BEEP
 .01,-10
 135 PRINT AT y+9,x;"  "
 140 PRINT AT y+8,x;"/\"
 145 IF y=6 THEN LET y=-9
 150 INK 3: LET j(y+12,x)=5: BEE
P .01,-10
 155 PRINT AT y+12,x;"  "
 165 PRINT AT y+11,x;"  "
 170 NEXT x
 300 LET p=5: LET q=15: LET r=5:
 PRINT AT 15,22;"    ";AT 17,5;
INK 1;"1    2    3"
 305 LET a1=0: LET a2=0: LET a3=
0
 310 IF j(9,5)=j(9,10) AND j(9,1
0)=j(9,15) THEN GO SUB 1000
 320 IF INT (RND*3)<2 THEN GO TO
 380
 325 BEEP .05,10: PRINT FLASH 1;
AT 15,22;"HOLD"
 335 IF INKEY$="0" THEN GO TO 80
0
 340 IF INKEY$="9" THEN LET a1=0
: LET a2=0: LET a3=0: PRINT AT 1
7,5; INK 1;"1    2    3"
 345 IF INKEY$="1" THEN LET a1=1
: PRINT AT 17,5; INK 1;"H": BEEP
 .05,20: PAUSE 20
 350 IF INKEY$="2" THEN LET a2=1
: PRINT AT 17,10; INK 1;"H": BEE
P .05,27: PAUSE 20
 355 IF INKEY$="3" THEN LET a3=1
: PRINT AT 17,15; INK 1;"H": BEE
P .05.34: PAUSE 20
 360 GO TO 335
 380 IF INKEY$="0" THEN GO TO 40
 400 GO TO 380
 800 IF a1=1 AND a2=1 AND a3=1 T
HEN BEEP .0125,40: BEEP .0125,40
: BEEP .0125,40: GO TO 300
 805 IF a1=1 AND a2=1 THEN LET p
=15: GO TO 40
 810 IF a2=1 AND a3=1 THEN LET q
=5: GO TO 40
 815 IF a1=1 AND a3=1 THEN LET p
=10: LET q=10: GO TO 40
 820 IF a1=1 THEN LET p=10: GO T
O 40
 825 IF a2=1 THEN LET r=10: GO T
O 40
 830 IF a3=1 THEN LET q=10: GO T
O 40
 835 GO TO 40
1000 FOR n=7 TO 2 STEP -1: BORDE
R n: PAUSE 10: NEXT n: FOR n=-7
TO 7 STEP .5: BEEP .02,49-n*n: N
EXT n
1005 FOR n=0 TO 20: PRINT INK 2;
AT n,28;"O O": BEEP .0125,40: NE
XT n
1010 IF j(9,5)=1 THEN LET sc=sc+
200
1020 IF j(9,5)=2 THEN LET sc=sc+
150
1030 IF j(9,5)=3 THEN LET sc=sc+
120
1040 IF j(9,5)=4 THEN LET sc=sc+
100
1050 IF j(9,5)=5 THEN LET sc=sc+
50
1060 PRINT AT 9,20;"      ";AT 9,
20;sc: FOR n=0 TO 20: PRINT AT n
,28;"   ": NEXT n
1070 PRINT AT 15,22;"    "
1080 IF INT (RND*5)=3 THEN GO SU
B 2000
1090 RETURN
2000 FOR n=1 TO 10: BORDER RND*7
: PAUSE 10: NEXT n: BORDER 2: PR
INT FLASH 1;EXP 2;AT 21,10;"JACK
POT": FOR o=1 TO 3: FOR w=20 TO
30: BEEP .0125,w: NEXT w: FOR v=
30 TO 25 STEP -1: BEEP .125,v: N
EXT v: NEXT o: FOR n=0 TO 20: PR
INT EXP 2;AT n,28;"ooo": BEEP .0
125,40: NEXT n: PAUSE 20: FOR n=
0 TO 20: PRINT AT n,28;"   ": NE
XT n: LET sc=sc+1000: PRINT AT 9
,20;"          ";AT 9,20;sc: PAUSE 5
0: PRINT AT 21,10;"          ": RET
URN
3000 DATA 255,255,255,127,127,63
,31,7,255,255,255,254,254,252,24
8,224,7,31,63,127,127,255,255,25
5,224,248,252,254,254,255,255,25
5,0,0,0,0,60,126,255,255
3010 DATA 31,63,63,63,63,31,31,7
,248,252,252,252,252,248,248,224
,1,1,1,3,3,7,31,31,128,128,128,1
92,192,224,240,248
3020 DATA 126,255,255,255,255,12
6,60,0,1,2,2,4,4,8,8,60,128,64,6
4,32,32,16,16,60
3030 DATA 9,31,9,1,3,1,0,0,32,24
0,32,0,128,0,0,0,1,1,73,255,73,7
3,255,73,0,0,36,254,36,36,254,36
```

NOTA:

```
   - GRAPHICS tecla A+B
   -    "    "   E+E
   -    "    "   C+D
   -    "    "   F+G
   -    "    "   H+I
   -    "    "   J+L
/\ -    "    "   K+L
   -    "    "   M+N
   -    "    "   O+P
```

FICHEIRO UNIVERSAL

SPECTRUM 48 K

Autor: ANTÓNIO JOÃO G. NUNES
FUNCHAL

Este programa, destinado à criação e tratamento de ficheiros, altera, lista, imprime, grava o ficheiro ou carrega outro já existente; Pesquisa por qualquer palavra. O programa leva apenas 20 segundo a carregar de cassete. Comprimento máximo dum ficheiro: 38500 bytes; dum registo: 275; dum campo: 275. Aplicações: agenda telefónica e de endereços, índice bibliotecário, catálogo de colecções, bloco de notas, etc.

Os passos para a introdução do programa são os seguintes:

1 — Introduza a linha 1: 1 REM 12345678901234

2 — Com um comando do tipo FOR N − 23760 TO 23773: INPUT F: POKE N,F: NEXT N, introduzir os números listados na fig. 1. Trata-se da rotina de carregamento dos bytes sem label, "disassemblada"na fig. 2.

3 — Transformar a linha 1 em linha 0 com o comando POKE 23756,0 para evitar o seu apagamento acidental.

4 — Introduzir o programa em BASIC listado na fig. 3.

5 — Gravá-lo com os comandos CLEAR: SAVE "FICH. UNIV3" LINE 20 (Não faça RUN pois ainda falta introduzir o código máquina).

6 — Com um comando do tipo FOR N =65032 TO 65535: INPUT F:POKE N, F: NEXT N dar entrada aos valores listados na fig. 4 Trata-se do código máquina correspondente às rotinas "disassembladas" na fig. 5.

7 — Gravar este bytes **sem label** logo a seguir ao programa em BASIC com o comando SAVE "mc" CODE 65032,504 só ligando o gravador depois de aparecerem as primeiras riscas azuis e amarelas (a fim de não gravar a label ou header).

8 — Carregar todo o programa com LOAD"" e testá-lo por exemplo com nomes e endereços de pessoas amigas, e verificar a rapidez com que é efectuada a pesquisa por qualquer palavra que conste do ficheiro.

O autor do programa utiliza-o para ficheiros de endereços e ainda para ficheiros de títulos de artigos de revistas de electrónica, com óptimos resultados.

O autor realça que o programa foi escrito com a principal preocupação de poupar memória a fim de a deixar disponível para o ficheiro, pelo que algumas instruções poderão parecer menos óbvias, como por exemplo LET Z = NOT PI que substitui LET Z = 0 economizando 5 bytes. Chama ainda a atenção para a instrução POKE 23658,8 que se destina a activar automaticamente o CAPS LOCK.

Quaisquer sugestões relativas a possíveis melhoramentos ao programa serão benvindas. *(António Nunes — Rua do Til, 72 — 9000 FUNCHAL)*

FIGURA 1

```
55,62,255,17,248,1,221,33,8,254,
205,86,5,201,
```

FIGURA 2

```
Rotina "LOAD-BYTES"

23760   scf
23761   ld a,255
23763   ld de,504
23766   ld ix,65032
23770   call 1366
23773   ret
```

FIGURA 3

```
  20 CLEAR VAL "65023": RANDOMIZ
E USR VAL "23760": LET F=VAL "38
501": LET N=PI/PI: DIM A$(VAL "3
1"): DIM B$(F): LET B$(N)=" STOP
 ": POKE VAL "23609",VAL "20": P
OKE VAL "23658",VAL "8": LET Z=N
OT PI: PAPER Z: BORDER Z: LET S=
VAL "6": LET U=N: INK S+U: LET K
=VAL "256"
  90 LET G=U
 130 CLS
 140 PRINT : PRINT ;"TEM ";F-N;"
 POSICOES LIVRES"''"OPCOES:"'"1.
 INTRODUZIR UM REGISTO"'"2. CONS
ULTAR O FICHEIRO"'"3. LISTAR/APA
GAR REGISTOS"'"4. GRAVAR O FICHE
IRO"'"5. CARREGAR UM FICHEIRO"'"
6. ACABAR"
 150 INPUT M: IF M<U OR M>S THEN
 GO TO VAL "150"
 170 CLS : GO TO VAL "1E3"*M
1000 INPUT "INTRODUZA UM REGISTO
","# PARA SEPARAR CAMPOS"'' LINE
 N$: IF N$="" THEN GO TO VAL "13
0"
1070 LET N$=N$+" STOP ": LET L=L
EN N$: IF N+L>F THEN PRINT FLASH
 U;"FALTA DE ESPACO": GO TO VAL
"140"
1110 LET B$(N+U TO N+L)=N$: LET
N=N+L: LET G=Z: CLS : PRINT "REG
ISTO INTRODUZIDO": GO TO VAL "1E
3"
2000 INPUT "INTRODUZA A CHAVE DE
 PESQUISA"'' LINE N$: IF N$="" T
HEN GO TO VAL "130"
2030 CLS : LET L=LEN N$: IF L>VA
L "30" THEN LET L=VAL "30"
2070 LET A$=N$( TO L)+" STOP ":
POKE VAL "65332",Z: POKE VAL "65
333",Z
2150 PRINT AT Z,Z;: LET P=USR VA
L "65336": IF P>Z AND P<=N THEN
GO TO VAL "2400"
2210 CLS : PRINT FLASH U;"FIM DO
 FICHEIRO": PRINT : GO TO VAL "2
E3"
2400 INPUT "1. EMENDAR","4. APAG
AR"'"2. IMPRIMIR"'"3. NOVA CHAVE
   ENTER.CONTINUAR"; LINE S$: IF
 S$="4" THEN GO SUB VAL "7E3": G
O TO VAL "2E3"
2410 IF S$="2" THEN GO SUB VAL "
8E3": GO TO VAL "2400"
2420 IF S$="3" THEN GO TO VAL "2
E3"
2425 IF S$="1" THEN GO TO VAL "2
500"
2430 CLS : GO TO VAL "2150"
2500 LET P=N+PEEK VAL "23627"+K*
PEEK VAL "23628"+VAL "54": POKE
VAL "65026",P-K*INT (P/K): POKE
VAL "65027",INT (P/K): LET N=USR
 VAL "65032": LET G=Z: GO TO VAL
 "2E3"
3000 LET A$=" STOP STOP ": POKE
VAL "65332",U: POKE VAL "65333",
Z
3110 PRINT AT Z,Z;: LET P=USR VA
L "65336": IF P>Z AND P<=N THEN
GO TO VAL "3300"
3210 CLS : PRINT FLASH U;"FIM DO
 FICHEIRO": GO TO VAL "140"
```

```
3300 INPUT "1. APAGAR REGISTO"'"
2. SAIR"'"ENTER. CONTINUAR LISTA
GEM"' LINE S$: IF S$="1" THEN GO
 SUB VAL "7E3": GO TO VAL "3300"
3325 IF S$="" THEN CLS : GO TO V
AL "3110"
3370 GO TO VAL "130"
4000 INPUT "NOME DO FICHEIRO ATE
 10 CARACT."'' LINE N$: IF N$=""
 THEN GO TO VAL "130"
4060 SAVE N$( TO VAL "10")CODE P
EEK VAL "23627"+K*PEEK VAL "2362
8",N+VAL "55": GO TO VAL "90"
5000 LOAD ""CODE PEEK VAL "23627
"+K*PEEK VAL "23628": GO TO VAL
"90"
6000 IF G THEN STOP
6010 PRINT FLASH U;"ATENCAO: NAO
 GRAVOU O FICHEIRO": PRINT : LET
 G=U: GO TO VAL "140"
7000 LET L=USR VAL "65470": LET
N=N-L: LET G=Z: RETURN
8000 FOR L=PEEK VAL "65334"+K*PE
EK VAL "65335" TO P: IF B$(L)="#
" OR B$(L)=" STOP " THEN LPRINT
: GO TO VAL "8020"
8010 LPRINT B$(L);
8020 NEXT L: RETURN
```

FIGURA 4

```
42,75,92,17,54,0,25,229,237,91,5
2,255,25,34,4,254,237,91,54,255,
225,25,34,6,254,42,4,254,34,0,25
4,205,26,255,42,4,254,54,143,205
,12,255,205,142,2,14,0,205,30,3,
56,246,205,142,2,14,0,205,30,3,4
8,246,21,95,205,51,3,245,205,142
,2,14,0,205,30,3,48,230,21,95,20
5,51,3,71,241,184,32,220,245,17,
10,0,33,250,0,205,181,3,241,254,
8,40,35,254,9,40,49,254,12,40,64
,254,13,40,105,245,205,26,255,42
,4,254,35,34,4,254,42,0,254,35,3
4,0,254,241,43,119,24,157,42,0,2
54,43,126,254,226,40,7,54,143,34
,0,254,35,119,24,139,42,0,254,35
,126,254,226,40,7,54,143,34,0,25
4,43,119,195,47,254,42,2,254,237
,91,0,254,27,167,237,82,68,77,42
,0,254,84,93,27,237,176,42,2,254
,43,34,2,254,42,4,254,43,34,4,25
4,42,0,254,43,34,0,254,195,47,25
4,42,2,254,237,91,0,254,167,237,
82,68,77,98,107,35,237,176,42,2,
254,43,34,2,254,237,91,75,92,167
,237,82,17,54,0,167,237,82,68,77
,201,205,107,13,42,6,254,35,126,
254,226,200,215,24,248,42,2,254,
35,34,2,254,237,91,0,254,167,237
,82,68,77,237,91,2,254,98,107,43
,237,184,201,0,0,248,0,221,42,75
,92,17,18,0,221,25,42,75,92,17,5
0,0,25,94,35,86,235,237,91,52,25
5,167,237,82,68,77,42,75,92,25,1
7,55,0,25,221,126,0,237,177,40,1
,201,221,229,229,221,35,221,126,
0,254,226,32,66,225,221,225,62,2
26,43,43,190,32,252,229,42,75,92
,17,54,0,25,235,225,167,237,82,3
4,54,255,25,35,126,254,226,40,14
,254,35,40,3,215,24,243,62,13,21
5,62,13,24,246,229,42,75,92,17,5
4,0,25,235,225,167,237,82,34,52,
255,68,77,201,190,32,3,35,24,175
,225,221,225,24,159,42,75,92,17,
53,0,25,94,35,86,42,52,255,235,1
67,237,82,68,77,42,75,92,17,55,0
,25,237,91,54,255,25,229,42,75,9
2,17,55,0,25,237,91,52,255,25,20
9,237,176,42,52,255,237,91,54,25
5,167,237,82,68,77,42,54,255,34,
52,255,201,
```

(CHECKSUM=55222)

FIGURA 5

```
  1   ROTINA "EDITOR"
  2
 10   ORG 65032
 20  ENDCURSOR EQU 65024
 30  FIMBLOCO EQU 65026
 40  STOPFIM EQU 65028
 50  STOPIN EQU 65030
 56
 57   LD HL,(23627)
 58   LD DE,54
 59   ADD HL,DE
 60   PUSH HL
 61   LD DE,(65332)
 62   ADD HL,DE
 63   LD (STOPFIM),HL
 64   LD DE,(65334)
 65   POP HL
 66   ADD HL,DE
 67   LD (STOPIN),HL
 70   LD HL,(STOPFIM)
 80   LD (ENDCURSOR),HL
 90   CALL MOVER
100   LD HL,(STOPFIM)
110   LD (HL),143
120  CONTINUA CALL ESCREVER
130  WAIT CALL #28E
140   LD C,0
150   CALL #31E
160   JR C,WAIT
170  NOKEY CALL #28E
180   LD C,0
190   CALL #31E
200   JR NC,NOKEY
210   DEC D
220   LD E,A
230   CALL #333
235   PUSH AF
236   CALL #28E
237   LD C,0
238   CALL #31E
239   JR NC,NOKEY
240   DEC D
241   LD E,A
242   CALL #333
243   LD B,A
244   POP AF
245   CP B
246   JR NZ,NOKEY
247   PUSH AF
249   LD DE,10
250   LD HL,250
260   CALL #3B5
270   POP AF
280   CP 8
290   JR Z,ESQUERDA
300   CP 9
310   JR Z,DIREITA
320   CP 12
330   JR Z,DELETAR
340   CP 13
350   JR Z,TERMINA
360   PUSH AF
370   CALL MOVER
380   LD HL,(STOPFIM)
390   INC HL
400   LD (STOPFIM),HL
410   LD HL,(ENDCURSOR)
420   INC HL
430   LD (ENDCURSOR),HL
440   POP AF
450   DEC HL
460   LD (HL),A
470   JR CONTINUA
480  ESQUERDA LD HL,(ENDCURSOR)
```

```
490  DEC HL
500  LD A,(HL)
510  CP 226
520  JR Z,FIMESQ
540  LD (HL),143
550  LD (ENDCURSOR),HL
560  INC HL
570  LD (HL),A
580 FIMESQ JR CONTINUA
590 DIREITA LD HL,(ENDCURSOR)
600  INC HL
610  LD A,(HL)
620  CP 226
630  JR Z,FIMDIR
640  LD (HL),143
650  LD (ENDCURSOR),HL
660  DEC HL
670  LD (HL),A
680 FIMDIR JP CONTINUA
690 DELETAR LD HL,(FIMBLOCO)
700  LD DE,(ENDCURSOR)
710  DEC DE
720  AND A
730  SBC HL,DE
740  LD B,H
750  LD C,L
760  LD HL,(ENDCURSOR)
770  LD D,H
780  LD E,L
790  DEC DE
800  LDIR
810  LD HL,(FIMBLOCO)
820  DEC HL
830  LD (FIMBLOCO),HL
840  LD HL,(STOPFIM)
850  DEC HL
860  LD (STOPFIM),HL
870  LD HL,(ENDCURSOR)
880  DEC HL
890  LD (ENDCURSOR),HL
900  JP CONTINUA
910 TERMINA LD HL,(FIMBLOCO)
920  LD DE,(ENDCURSOR)
930  AND A
940  SBC HL,DE
950  LD B,H
960  LD C,L
970  LD H,D
980  LD L,E
990  INC HL
1000 LDIR
1010 LD HL,(FIMBLOCO)
1020 DEC HL
1030 LD (FIMBLOCO),HL
1040 LD DE,(23627)
1050 AND A
1060 SBC HL,DE
1070 LD DE,54
1080 AND A
1090 SBC HL,DE
1100 LD B,H
1110 LD C,L
1120 RET
1130 ESCREVER CALL #D6B
1140 LD HL,(STOPIN)
1150 LOOP INC HL
1160 LD A,(HL)
1170 CP 226
1180 RET Z
1190 RST 16
1200 JR LOOP
1210 MOVER LD HL,(FIMBLOCO)
1220 INC HL
1230 LD (FIMBLOCO),HL
1240 LD DE,(ENDCURSOR)
1250 AND A
1260 SBC HL,DE
1270 LD B,H
1280 LD C,L
1290 LD DE,(FIMBLOCO)
1300 LD H,D
1310 LD L,E
1320 DEC HL
1330 LDDR
1340 RET
 10 ROTINAS "PESQUISA" E
    "APAGA"
 11
 20  ORG 65336
 30 VARS EQU 23627
 40 STOPFIM EQU 65332
 50 STOP EQU 226
 60 FIMLINHA EQU 35
 70 STOPINI EQU 65334
 80
 90   LD IX,(VARS)
100   LD DE,18
110   ADD IX,DE
120   LD HL,(VARS)
130   LD DE,50
140   ADD HL,DE
150   LD E,(HL)
160   INC HL
170   LD D,(HL)
180   EX DE,HL
190   LD DE,(STOPFIM)
200   AND A
210   SBC HL,DE
220   LD B,H
230   LD C,L
240   LD HL,(VARS)
250   ADD HL,DE
260   LD DE,55
270   ADD HL,DE
280 RECOMECA  LD A,(IX+0)
290   CPIR
300   JR Z,ENCONTROU
310   RET
320 ENCONTROU PUSH IX
330   PUSH HL
340 IGUAIS    INC IX
350   LD A,(IX+0)
360   CP STOP
370   JR NZ,COMPARA
380   POP HL
390   POP IX
400   LD A,STOP
405   DEC HL
410 DECREMENTA DEC HL
420   CP (HL)
430   JR NZ,DECREMENTA
431   PUSH HL
432   LD HL,(VARS)
433   LD DE,54
434   ADD HL,DE
435   EX DE,HL
436   POP HL
437   AND A
438   SBC HL,DE
440   LD (STOPINI),HL
445   ADD HL,DE
450 CICLO   INC HL
460   LD A,(HL)
470   CP STOP
480   JR Z,ACABA
490   CP FIMLINHA
500   JR Z,NOVALINHA
510 ESCREVE  RST #10
520   JR CICLO
530 NOVALINHA LD A,13
532   RST #10
535   LD A,13
540   JR ESCREVE
550 ACABA  PUSH HL
560   LD HL,(VARS)
570   LD DE,54
580   ADD HL,DE
590   EX DE,HL
600   POP HL
620   AND A
630   SBC HL,DE
640   LD (STOPFIM),HL
643   LD B,H
646   LD C,L
```

```
650    RET
660 COMPARA   CP (HL)
670    JR NZ,DIFERENTE
680    INC HL
690    JR IGUAIS
700 DIFERENTE  POP HL
710    POP IX
720    JR RECOMECA
750 DELETE   LD HL,(VARS)
760    LD DE,53
770    ADD HL,DE
780    LD E,(HL)
790    INC HL
800    LD D,(HL)
810    LD HL,(STOPFIM)
820    EX DE,HL
830    AND A
840    SBC HL,DE
850    LD B,H
860    LD C,L
870    LD HL,(VARS)
880    LD DE,55
890    ADD HL,DE
900    LD DE,(STOPIN)
910    ADD HL,DE
920    PUSH HL
930    LD HL,(VARS)
940    LD DE,55
950    ADD HL,DE
960    LD DE,(STOPFIM)
970    ADD HL,DE
980    POP DE
990    LDIR
1000   LD HL,(STOPFIM)
1010   LD DE,(STOPIN)
1020   AND A
1030   SBC HL,DE
1040   LD B,H
1050   LD C,L
1060   LD HL,(STOPIN)
1070   LD (STOPFIM),HL
1080   RET
```

NOVOS COMANDOS DE EXECUÇÃO GRÁFICA

SPECTRUM 16/48 K

in. YOUR COMPUTER, Março 84
Adapt. e Trad.: J. MAGALHÃES

Para produzir gráficos, o Spectrum possui alguns comandos (PLOT, DRAW, CIRCLE) que, no entanto, são muito rudimentares para uma completa exploração do potencial gráfico de alta resolução.
Outras máquinas possuem comandos tais como Elipses, Rectangulos e Triangulos, com a facilidade de preencher áreas específicas de diferentes cores.
Este programa pretende remediar esse problema.

A INTRODUÇÃO DO CÓDIGO MÁQUINA

O Código Máquina tem uma extensão de 744 bytes, apresentados na listagem 1 para o Spectrum 48 K e na listagem 8 para o 16 K.
Para dar entrada do código, precisa de um programa "carregador" como o apresentado na listagem 2 (48 K) e na listagem 9 (16 K). Basta apenas passar um destes programas (conforme a máquina), fazer RUN e dar entrada dos bytes um a um.
É muito importante verificar se os números correspondem correctamente aos endereços apresentados.
Depois de carregados os 744 bytes, se verificar a mensagem de "ERRO", houve qualquer falta na introdução do C.M. Para localizar o erro e corrigi-lo, proceda do seguinte modo:

— Para o Spectrum 48 K passe a listagem 3.
— Para o Spectrum 16 K passe a listagem 10.

Depois de fazer RUN, será interrogado se está a usar a impressora. Respondendo afirmativamente (S), sairá na impressora o C.M. introduzido anteriormente. Verifique se corresponde ao da listagem que apresentámos. Para correcção de qualquer entrada faça:

POKE (endereço), (n.º correcto)
Ex.: POKE 65218, 201

Corrigidos todos os erros, o programa deverá estar pronto a funcionar.

PARA GRAVAR O CÓDIGO MÁQUINA:

SAVE "Gráficos" CODE 64624, 744 (Para 48 K)
SAVE "Gráficos" CODE 31856, 744 (Para 16 K)

PARA CARREGAR O CÓDIGO MÁQUINA:

CLEAR 64623 : LOAD " " CODE (Para 48 K)
CLEAR 31855 : LOAD " " CODE (Para 16 K)

OS "NOVOS" COMANDOS:

Serão usados em linhas REM (REM Statments), que devem ser precedidos das instruções:

RANDOMIZE USR 64624 (Para 48 K)
RANDOMIZE USR 31856 (Para 16 K)

Portanto, estas instruções deverão encontrar-se na mesma linha a seguir ao comando de chamada do código máquina (RANDOMIZE USR . . .) ou no início da linha seguinte, tal como apresentámos nos exemplos.
Os comandos poderão ser dados em minúsculas, maiúsculas ou numa combinação de ambas, dado que o Spectrum não terá problemas em reconhecer.
Qualquer erro será interpretado pelo computador como "ERROR C".

Quando se trabalha com instruções BASIC normais, usam-se os parâmetros de cor antes da palavra-chave. Ora isso não vai ser possível com estes novos comandos. Se pretender usar cores como INK, PAPER, FLASH, BRIGHT, OVER e INVERSE em "statments" isoladas, deve usar o comando TEMPS no início da primeira REM statment. Isso indica ao computador que deseja usar determinada cor em todos os comandos sucessivos da linha REM. Por outro lado, se essa cor tiver uma função temporária, faça uma linha PRINT a preceder a instrução RANDOMIZE USR:

PRINT, INK 3 : PAPER 7

- **MOVE x, y** — Tem uma aplicação idêntica ao PLOT normal (posição x, y) e é usado especialmente para colorir uma determinada área. Os parâmetros x, y podem ser dados de duas formas:

  — directamente MOVE 128, 88
  — usando variáveis MOVE a, b

  Repare, no entanto, que não pode dar entrada do comando MOVE a – 1, b/3. Terá de o substituir por LET x = a – 1 : LET y = b/3. E então MOVE x, y.

- **LINE x, y ou LINES (x1, y1) _ (x2, y2)** — Este comando pode ser dado sob duas formas:

  — LINE x, y

Onde x e y são as coordenadas de um ponto no écran, desenhando uma linha entre dois pontos. A vantagem desta instrução está no uso de maior número de pontos no écran do que o já conhecido comando DRAW.
A cor da linha é determinada por uma "PRINT statment", precedida do comando RANDOMIZE USR no caso de cores temporárias, ou precedida de uma instrução TEMPS no caso de uma cor universal.

— A segunda forma deste comando é idêntica ao comando LINE. A sua forma é LINE (x1, y1) – (x2, y2) e permite desenhar uma linha do ponto x1, y1, até ao ponto x2, y2. O efeito é o mesmo, embora menos prático, com MOVE x1, y1/LINE x2, y2.

- **BOX x, y** — Este comando desenha um rectângulo com um comprimento x e uma largura y. O último posicionamento será o do canto inferior esquerdo do rectângulo. Mais uma vez a cor é determinada da mesma forma que no comando LINE.
- **CIRCLE x, y** — De forma diferente do comando normal CIRCLE, este permite a construção de elipses e círculos. O centro do círculo é determinado pelo último posicionamento, sendo x o raio e y a razão altura/comprimento, que deve situar-se entre os limites 1 a 255, 10.
- **PAINT** — Este comando permite colorir determinada área do écran, a partir do último posicionamento, em todas as direcções até à linha que a delimita.

É importante que a posição inicial não seja um ponto em que já foi usada uma cor INK pois, nesse caso, o computador tomaria essa linha como limite, não permitindo o colorido com precisão.
A rotina não funcionará entre 2 objectos separados (ver fig. 2).

- **WCLS a . b . c . d** — Este comando permite limpar qualquer área do écran, facilitando a tarefa de usar diferentes parte do écran com diferentes gráficos.
Os parâmetros "a" e "b" referem-se à coluna e às posições da linha respectivamente do canto superior esquerdo da área a ser limpa.
O parâmetro "c" refere-se ao número de colunas a limpar e o parâmetro "d" indica o n.º de linhas (ver fig. 3).
A cor da área a ser limpa funciona da mesma maneira com este comando.
Repare, contudo, que quando usa a instrução TEMPS, a área é limpa com:

PAPER 8 : FLASH 8; BRIGHT 8;

Apenas a cor INK é alterada. Pode remediar esta situação fazendo um PRINT vazio. P. ex.: PRINT.
As listagens 4 e 7 exemplificam todos estes novos comandos e ajudam a compreender qualquer pormenor menos explícito. Estas listagens só funcionam no Spectrum 48 K. Para o 16 K terá de alterar as instruções RANDOMIZE USR 64624 para RANDOMIZE USR 31856.

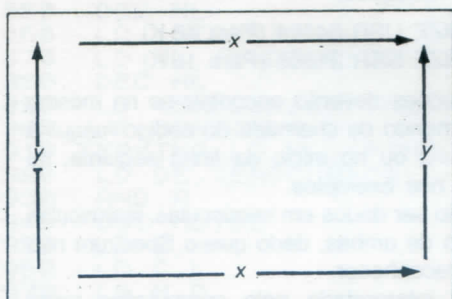



Figura 1

Figura 2

Figura 3

Listagem 1

```
64624 - 253 203 12  126 192 42
64630 - 93  92  229 42  69  92
64636 - 205 110 25  253 86  13
64642 - 20  30  0   205 136 25
64648 - 48  14  42  85  92  62
64654 - 192 166 32  11  35  35
64660 - 35  34  93  92  231 254
64666 - 234 40  2   207 11  231
64672 - 17  224 252 229 26  230
64678 - 127 79  223 246 32  185
64684 - 32  9   231 26  19  23
64690 - 48  240 225 24  17  26
64696 - 19  23  48  251 19  19
64702 - 26  225 34  93  92  167
64708 - 32  221 207 11  235 94
64714 - 35  86  235 205 223 252
64720 - 223 254 47  40  202 225
64726 - 34  93  92  217 33  88
64732 - 39  217 201 233 108 105
64738 - 110 229 74  253 98  111
64744 - 248 145 253 99  105 114
64750 - 99  108 229 50  254 112
64756 - 97  105 110 244 210 253
64762 - 109 111 118 229 199 253
64768 - 119 99  108 243 225 254
64774 - 116 101 109 112 243 180
64780 - 253 0   205 25  253 223
64786 - 254 44  40  2   207 11
64792 - 231 223 205 136 44  48
64798 - 247 205 141 44  56  18
64804 - 205 59  45  223 254 13
64810 - 200 254 44  200 254 47
64816 - 200 254 41  200 207 11
64822 - 229 231 205 39  253 225
64828 - 34  93  92  205 178 40
64834 - 56  4   35  195 180 51
64840 - 207 1   223 254 40  32
64846 - 39  231 205 14  253 205
64852 - 7   35  205 229 34  223
64858 - 254 41  32  22  231 254
64864 - 45  32  17  231 254 40
64870 - 32  12  231 205 14  253
64876 - 223 254 41  32  3   231
64882 - 24  5   207 11  205 14
64888 - 253 253 126 68  205 40
64894 - 45  253 126 67  205 40
64900 - 45  239 192 2   3   1
64906 - 224 3   1   56  195 183
64912 - 36  205 14  253 239 49
64918 - 160 1   56  205 183 36
64924 - 239 1   49  160 56  205
64930 - 183 36  239 1   27  160
64936 - 1   56  205 183 36  239
64942 - 27  160 56  195 183 36
64948 - 253 203 2   134 205 77
64954 - 13  33  144 92  126 246
64960 - 248 119 253 203 87  182
64966 - 201 205 14  253 205 7
64972 - 35  237 67  125 92  201
64978 - 237 75  125 92  80  89
```

```
64984 - 205 242 253 56  6   4
64990 - 120 254 176 56  245 66
64996 - 120 5   167 200 205 242
65002 - 253 216 120 5   167 32
65008 - 247 201 75  205 27  254
65014 - 48  6   121 187 55  200
65020 - 24  3   12  32  242 75
65026 - 121 13  167 55  63  200
65032 - 205 27  254 48  7   123
65038 - 61  185 55  200 183 201
65044 - 121 13  167 32  239 183
65050 - 201 197 213 205 170 34
65056 - 71  4   126 7   16  253
65062 - 56  7   121 230 7   205
65068 - 236 34  183 209 193 201
65074 - 253 126 67  205 40  45
65080 - 253 126 68  205 40  45
65086 - 205 14  253 239 164 5
65092 - 195 2   194 2   193 2
65098 - 192 2   160 226 227 4
65104 - 56  205 213 45  79  58
65110 - 125 92  129 50  125 92
65116 - 239 49  52  64  176 0
65122 - 90  5   163 4   49  56
65128 - 33  195 254 34  104 92
65134 - 239 32  56  33  146 92
65140 - 34  104 92  239 226 4
65146 - 227 4   224 15  1   56
65152 - 33  195 254 34  104 92
65158 - 239 31  56  33  146 92
65164 - 34  104 92  239 226 4
65170 - 225 15  56  33  195 254
65176 - 34  104 92  205 121 253
65182 - 33  146 92  34  104 92
65188 - 239 164 162 4   15  49
65194 - 56  205 162 45  33  104
65200 - 1   183 237 66  48  166
65206 - 239 2   224 225 56  205
65212 - 7   35  237 67  125 92
65218 - 201 0   0   135 0   0
65224 - 121 25  236 136 91  126
65230 - 155 154 95  35  0   0
65236 - 0   0   0   0   0   0
65242 - 0   0   0   0   0   0
65248 - 0   205 14  253 223 254
65254 - 44  40  2   207 11  231
65260 - 205 14  253 205 7   35
65266 - 197 205 7   35  209 213
65272 - 197 123 167 40  89  129
65278 - 56  86  254 33  48  82
65284 - 122 167 40  78  128 56
65290 - 75  254 23  48  71  209
65296 - 62  24  146 71  205 155
65302 - 14  22  0   25  193 203
65308 - 32  203 32  203 32  197
65314 - 229 54  0   229 35  235
65320 - 225 13  6   0   120 177
65326 - 40  2   237 176 225 193
65332 - 36  124 230 7   32  25
65338 - 37  229 197 229 205 219
65344 - 11  225 35  13  32  247
65350 - 193 225 36  125 198 32
65356 - 111 63  159 230 248 132
65362 - 103 16  204 201 207 10
```

Listagem 2

```
  10 CLEAR 64623
  20 LET total=0
  30 PRINT AT 21,0;
  40 FOR a=64624 TO 65367
  50 INPUT "Byte? ";b
  60 POKE a,b: LET total=total+b
  70 POKE 23692,-1
  80 PRINT a;" - ";PEEK a
  90 NEXT a
 100 IF total<>89072 THEN PRINT
FLASH 1; BRIGHT 1;"ERRO": BEEP 3
,-30
```

Listagem 3

```
  10 CLOSE #2: INPUT "COM IMPRES
SORA? ";a$
  20 IF a$(1)="S" OR a$(1)="s" T
HEN OPEN #2,"p"
  30 LET a=64624
  40 PRINT a;" - ";
  50 FOR b=0 TO 5
  60 PRINT TAB 8+b*4;PEEK (a+b);
  70 NEXT b
  80 LET a=a+6
  90 IF a<65368 THEN PRINT : GO
TO 40
 100 CLOSE #2
```

Listagem 4

```
  10 FOR A=0 TO 255 STEP 8
  20 RANDOMIZE USR 64624: REM TE
MPS/LINE(0,0)-(A,175)
  30 NEXT A
  40 FOR A=0 TO 255 STEP 5
  50 RANDOMIZE USR 64624: REM TE
MPS/LINE (A,0)-(A,175)
  60 NEXT A
```

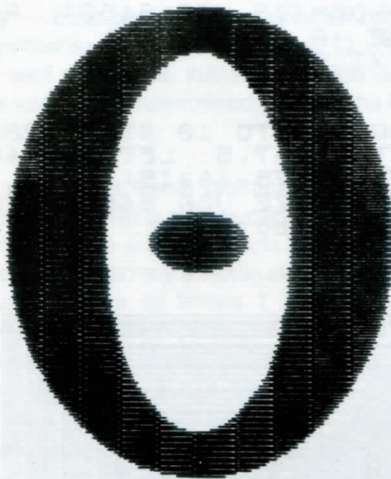
Listagem 5

```
  10 FOR a=0 TO 10 STEP .25
  20 LET b=a*7.5: LET d=b+40
  30 LET c=175-(a*15)
  40 RANDOMIZE USR 64624: REM te
mps/moved,b/boxc,c
  50 NEXT a
```

Listagem 6

```
  10 FOR a=1 TO 18
  20 RANDOMIZE USR 64624
  30 REM temps/move128,88/circle
70,a
  40 NEXT a
```

Listagem 7

```
  10 RANDOMIZE USR 64624: REM te
mps/move128,88/circle10,20
  20 RANDOMIZE USR 64624: REM te
mps/move128,88/circle65,6
  30 CIRCLE 128,88,80
  40 PRINT INK 2;
  50 RANDOMIZE USR 64624
  60 REM move128,88/paint
  70 PRINT INK 1;
  80 RANDOMIZE USR 64624
  90 REM move85,88/paint/move172
,88/paint/move128,20/paint/move1
28,155/paint
```

Listagem 8

```
31856 - 253 203 12 126 192 42
31862 - 93 92 229 42 69 92
31868 - 205 110 25 253 86 13
31874 - 20 30 0 205 136 25
31880 - 48 14 42 85 92 62
31886 - 192 166 32 11 35 35
31892 - 35 34 93 92 231 254
31898 - 234 40 2 207 11 231
31904 - 17 224 124 229 26 230
31910 - 127 79 223 246 32 185
31916 - 32 9 231 26 19 23
31922 - 48 240 225 24 17 26
31928 - 19 23 48 251 19 19
31934 - 26 225 34 93 92 167
31940 - 32 221 207 11 235 94
31946 - 35 86 235 205 223 124
31952 - 223 254 47 40 202 225
31958 - 34 93 92 217 33 88
31964 - 39 217 201 233 108 105
31970 - 110 229 74 125 98 111
31976 - 248 145 125 99 105 114
31982 - 99 108 229 50 126 112
31988 - 97 105 110 244 210 125
31994 - 109 111 118 229 199 125
32000 - 119 99 108 243 225 126
32006 - 116 101 109 112 243 180
32012 - 125 0 205 25 125 223
32018 - 254 44 40 2 207 11
32024 - 231 223 205 136 44 48
32030 - 247 205 141 44 56 18
32036 - 205 59 45 223 254 13
32042 - 200 254 44 200 254 47
32048 - 200 254 41 200 207 11
32054 - 229 231 205 39 125 225
32060 - 34 93 92 205 178 40
32066 - 56 4 35 195 180 51
32072 - 207 1 223 254 40 32
32078 - 39 231 205 14 125 205
32084 - 7 35 205 229 34 223
32090 - 254 41 32 22 231 254
32096 - 45 32 17 231 254 40
32102 - 32 12 231 205 14 125
32108 - 223 254 41 32 3 231
32114 - 24 5 207 11 205 14
32120 - 125 253 126 68 205 40
32126 - 45 253 126 67 205 40
32132 - 45 239 192 2 3 1
32138 - 224 3 1 56 195 183
32144 - 36 205 14 125 239 49
32150 - 160 1 56 205 183 36
32156 - 239 1 49 160 56 205
32162 - 183 36 239 1 27 160
32168 - 1 56 205 183 36 239
32174 - 27 160 56 195 183 36
32180 - 253 203 2 134 205 77
32186 - 13 33 144 92 126 246
32192 - 248 119 253 203 87 182
32198 - 201 205 14 125 205 7
32204 - 35 237 67 125 92 201
32210 - 237 75 125 92 80 89
32216 - 205 242 125 56 6 4
32222 - 120 254 176 56 245 66
32228 - 120 5 167 200 205 242
32234 - 125 216 120 5 167 32
32240 - 247 201 75 205 27 126
32246 - 48 6 121 187 55 200
32252 - 24 3 12 32 242 75
32258 - 121 13 167 55 63 200
32264 - 205 27 126 48 7 123
32270 - 61 185 55 200 183 201
32276 - 121 13 167 32 239 183
32282 - 201 197 213 205 170 34
32288 - 71 4 126 7 16 253
32294 - 56 7 121 230 7 205
32300 - 236 34 183 209 193 201
32306 - 253 126 67 205 40 45
32312 - 253 126 68 205 40 45
32318 - 205 14 125 239 164 5
32324 - 195 2 194 2 193 2
32330 - 192 2 160 226 227 4
32336 - 56 205 213 45 79 58
32342 - 125 92 129 50 125 92
32348 - 239 49 52 64 176 0
32354 - 90 5 163 4 49 56
32360 - 33 195 126 34 104 92
32366 - 239 32 56 33 146 92
32372 - 34 104 92 239 226 4
32378 - 227 4 224 15 1 56
32384 - 33 195 126 34 104 92
32390 - 239 31 56 33 146 92
32396 - 34 104 92 239 226 4
32402 - 225 15 56 33 195 126
32408 - 34 104 92 205 121 125
32414 - 33 146 92 34 104 92
32420 - 239 164 162 4 15 49
32426 - 56 205 162 45 33 104
32432 - 1 183 237 66 48 166
32438 - 239 2 224 225 56 205
32444 - 7 35 237 67 125 92
32450 - 201 0 0 0 0 0
32456 - 0 0 0 0 0 0
32462 - 0 0 0 0 0 0
32468 - 0 0 0 0 0 0
32474 - 0 0 0 0 0 0
32480 - 0 205 14 125 223 254
32486 - 44 40 2 207 11 231
32492 - 205 14 125 205 7 35
32498 - 197 205 7 35 209 213
32504 - 197 123 167 40 89 129
32510 - 56 86 254 33 48 82
32516 - 122 167 40 78 128 56
32522 - 75 254 23 48 71 209
32528 - 62 24 146 71 205 155
32534 - 14 22 0 25 193 203
32540 - 32 203 32 203 32 197
32546 - 229 54 0 229 35 235
32552 - 225 13 6 0 120 177
32558 - 40 2 237 176 225 193
32564 - 36 124 230 7 32 25
32570 - 37 229 197 229 205 219
32576 - 11 225 35 13 32 247
32582 - 193 225 36 125 198 32
32588 - 111 63 159 230 248 132
32594 - 103 16 204 201 207 10
```

Listagem 9

```
  10 CLEAR 31855
  20 LET total=0
  30 PRINT AT 21,0;
  40 FOR a=31856 TO 32599
  50 INPUT "Byte? ";b
  60 POKE a,b: LET total=total+b
  70 POKE 23692,-1
  80 PRINT a;" - ";PEEK a
  90 NEXT a
 100 IF total<>84189 THEN PRINT 
FLASH 1; BRIGHT 1;"ERRO": BEEP 3
,-30
```

Listagem 10

```
  10 CLOSE #2: INPUT "COM IMPRES
SORA? ";a$
  20 IF a$(1)="S" OR a$(1)="s" T
HEN OPEN #2,"p"
  30 LET a=31855
  40 PRINT a;" - ";
  50 FOR b=0 TO 5
  60 PRINT TAB 8+b*4;PEEK (a+b);
  70 NEXT b
  80 LET a=a+6
  90 IF a<32600 THEN PRINT : GO
TO 40
 100 CLOSE #2
```

HEADER

SPECTRUM

Autor: PAULO CRUZ
LISBOA

"Envio uma listagem de um programa que possuo, algo semelhante ao HEADER que consta na revista do mês de Março (pág. 14) que penso possuir algumas vantagens em relação àquele". *(Paulo Cruz)*

```
  10 CLEAR 32511
  20 FOR a=32512 TO 32521: READ
b: POKE a,b: NEXT a
  30 DATA 175,55,221,33,16,127,2
05,86,5,201
  40 LET b=32528: DEF FN a(x)=PE
EK (b+x)+256*PEEK (b+x+1)
  45 PRINT PAPER 2; INK 7; FLASH
 1;AT 11,10;" LOADING... ": PLOT
 78,78: DRAW 100,0: DRAW 0,11: D
RAW -100,0: DRAW 0,-11
  50 RANDOMIZE USR 32512
  60 LET c=PEEK b
  70 IF c>3 THEN GO TO 45
  75 BEEP .5,4: CLS
  80 PRINT '"NOME..............>
";
  90 FOR a=b+1 TO b+10: PRINT FL
ASH 1; PAPER 6;CHR$ PEEK a;: NEX
T a
 100 PRINT AT 3,0;"TIPO.........
......> "
 110 GO SUB 1000+100*c
 125 POKE b,255
 130 GO TO 45
1000 PRINT AT 3,20;"BASIC"
```

```
1010 PRINT AT 5,0;"COMP. TOTAL.
...> ";FN a(11);" bytes"
1020 PRINT AT 7,0;"COMP. PROGRA
MA..> ";FN a(15);" bytes"
1040 PRINT AT 9,0;"RUNS DA LINH
A...> "
1042 IF FN a(13)>9999 THEN PRINT
 AT 9,20;"SO LOAD """: RETURN
1045 PRINT AT 9,20;FN a(13)
1050 RETURN
1100 PRINT AT 3,20;"ARRAY NUMERI
CO"
1110 LET a$="": GO TO 1220
1200 PRINT AT 3,20;"ARRAY DE CAR
ACTERES"
1210 LET a$="$"
1220 PRINT AT 5,0;"COMP. DO ARR
AY..> ";FN a(11);" bytes"
1230 LET d=PEEK (b+14)
1240 PRINT AT 7,0;"NOME/ORIG. A
RRAY.> ";CHR$ (64+32*(d/32-INT (
d/32)));a$
1250 RETURN
1300 IF FN a(11)=6912 AND FN a(1
3)=16384 THEN PRINT AT 3,20;"SCR
EEN $": RETURN
1310 PRINT AT 3,20;"COD. MAQ."
1320 PRINT AT 5,0;"END. DE INIC
IO..> Byte ";FN a(13)
1330 PRINT AT 7,0;"COMP. PROGRA
MA..> ";FN a(11);" bytes"
1340 RETURN
1350 REM PARA GRAVAR
1360 SAVE "BYTE-SPY" LINE 10
1370 CLS : PRINT FLASH 1;" QUERO
 VERIFICAR ISTO..."
1380 VERIFY ""
```

PROGRAMA DE CÓPIA

SPECTRUM 16/48 K

Adapt.: CLUBE Z80

Esta rotina em código máquina permite fazer a cópia de quase todos os programas 16 ou 48 K / Spectrum.
Introduza o programa tal como na listagem e faça RUN. Quando o programa estiver pronto a ser gravado (mensagem "start tape, then press any Key"), prepare uma cassete limpa e inicie a gravação.

PARA CARREGAR O PROGRAMA:

LOAD "CÓPIA" CODE ou apenas LOAD " " CODE

Depois de carregado:

RANDOMIZE USR 23296

Segue-se um pequeno menu de opções:

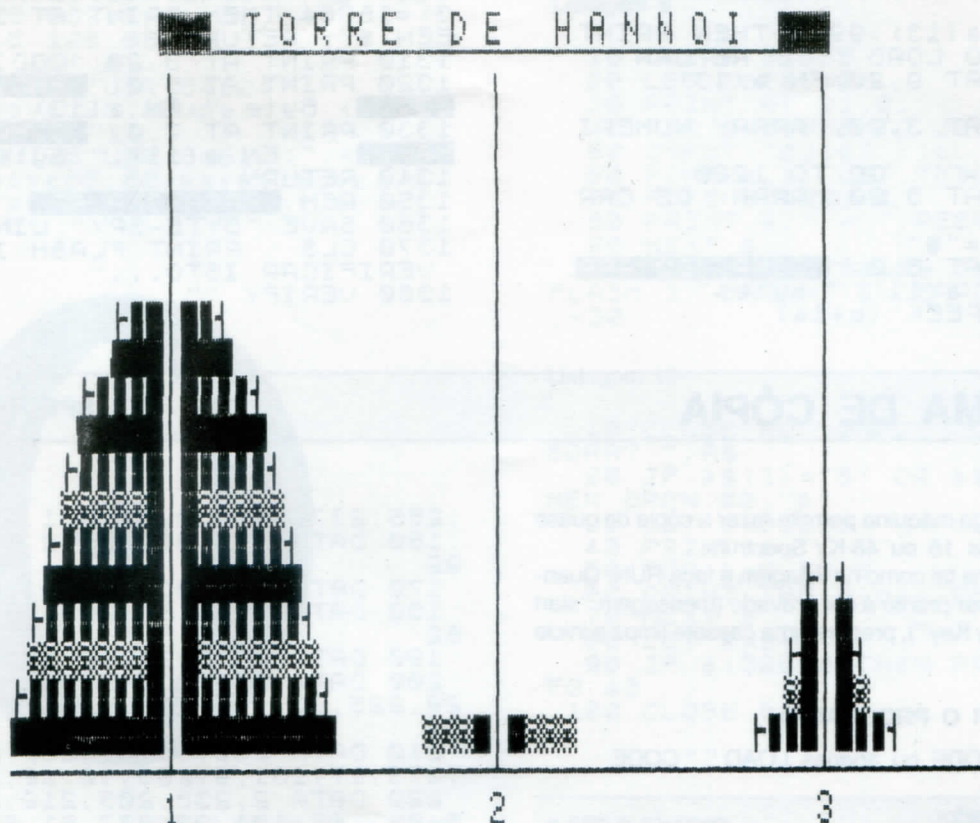
1 — Blocos normais (com header)
2 — Blocos sem header

Conforme o programa que quiser copiar, escolha uma das opções. Coloque o cabo de gravação no EAR do Spectrum e no EAR do gravador e comece a passar o programa que quer copiar. Repare que só pode passar e gravar um bloco de cada vez.
Depois de gravado, terá de novo as opções iniciais para dar entrada de um novo bloco, se existir.
A opção 3 permite a passagem ao BASIC.

```
   5 LET a=0
   6 RESTORE 50
  10 FOR i=23296 TO 23296+238
  20 READ c: POKE i,c: LET a=a+c
  30 NEXT i
  35 PRINT ''a
  40 IF a<>22885 THEN PRINT '"ER
RO": STOP
  45 SAVE "COPIA"CODE 23296,239
  50 DATA 62,2,205,1,22,6,22,205
,68,14,17,184,91,1,55,0
  70 DATA 205,60,32,33,0,91,229,
62,247,219,254,79,230
  90 DATA 1,40,12,121,230,2,40,6
6,121,230,4,32,237,225,201
 110 DATA 62,49,205,166,91,175,5
5,221,33,239,91,17,17,0
 120 DATA 205,2,8,58,239,91,17,1
92,9,205,10,12,1,10,0,17
 150 DATA 240,91,205,60,32,55,62
,255,237,91,250,91,221,42
 160 DATA 101,92,205,2,8,42,101,
92
 170 DATA 221,33,239,91,195,112
 180 DATA 9,62,50,205,166,91,55,
62
 190 DATA 255,221,42,101,92,17
 200 DATA 255,239,205,86,5,221,2
29,225,237,91,101,92,175,237,82,
34
 210 DATA 250,91,62,253,205,1,22
,175,17,161,9,205,10,12,253,203
 220 DATA 2,238,205,212,21,62,25
5,221,42,101,92,237,91,250,91,19
5
 230 DATA 194,4,215,1,8,0,17,176
,91,195,60,32
 240 DATA 22,1,0,6,6,22,0,0,22,3
,0,127,74,80,39,56,52,13,49,45
 250 DATA 78,79,82,77,65,76,32,6
7,79,80,89,13,50,45,72,69,65,68,
69,82,76,69,83,83
 260 DATA 32,67,79,80,89,13,51,4
5,69,78
 270 DATA 68,13,79,80,84,73,79,7
8,63
```

TORRE DE HANNOI

NEWBRAIN

Autor: PAULO CASTELO
PORTO

```
10 REM
20 REM    "*12:HN"    VERSAO 1.ii    5/6/84
30 REM
40 REM
50 REM    NB840605I          Paulo Castelo
60 REM
70 REM    NEWBRAIN AD  -  ANSI BASIC PLUS
80 REM
1100 OPEN#0,0,"L30" : CLOSE#5 : OPEN#5,5 : PUT#5, 1 : DIM P(3,18), P$(18)
1110 PRINT TAB(21);"███  T O R R E   D E   H A N N O I  ███"
1120 PRINT TAB(21);"________________________________________" : PUT 22,12,21
1130 PRINT"___________________________________________________"
1140 PRINT TAB(21);"1"; TAB(40);"2"; TAB(59);"3"
1150 GOSUB 2000
1300 PUT 22,18,24 : PRINT "TRANSFERE UM ANEL DA TORRE .█. PARA A TORRE .█."
1310 PUT 22,46,24,6 : GET#5, A : IF (A<49 OR A>51) AND A<>32 THEN 1310
1320 IF A<>32 GOTO 1350
1330 PUT 13,30 : PRINT TAB(18);"DESISTE ? █";CHR$(8); : GET#5, A : IF A=83 PUT A
█,12 : GOTO 1480
1340 PUT 30 : GOTO 1300
1350 PUT A : IF PN(A-48)=0 PUT 13,30 : PRINT TAB(29);"NAO HA ANEIS NA TORRE .";C
█HR$(A);"."; : GOTO 1300
1360 PUT 22,63,24,6 : GET#5, B : IF (B<49 OR B>51) AND B<>32 THEN 1360
1370 IF B<>32 GOTO 1400
1380 PUT 13,30 : PRINT TAB(18);"DESISTE ? █";CHR$(8); : GET#5, A : IF A=83 PUT A
█,12 : GOTO 1480
1390 PUT 30 : GOTO 1300
1400 IF A=B PUT 13,30 : PRINT TAB(25);"NAO PODE TROCAR COM A MESMA TORRE"; : GOT
█O 1300
1410 A=A-48 : B=B-48 : PUT 7
1420 IF P(B,PN(B))<P(A,PN(A)) AND PN(B)<>0 PUT 13,30 : PRINT TAB(18);"NAO PODE P
█OR UM ANEL MAIOR EM CIMA DE UM MENOR"; : GOTO 1300
1430 PN(B)=PN(B)+1 : P(B,PN(B))=P(A,PN(A)) : P(A,PN(A))=0
1440 X=12+19*(A-1) : Y=21-PN(A) : PUT 22,X,Y : PRINT P$(0); : PN(A)=PN(A)-1
1450 X=12+19*(B-1) : Y=21-PN(B) : PUT 22,X,Y : PRINT P$(P(B,PN(B)));
```

```
1460 IF PN(2)<>18 AND PN(3)<>18 PUT 22,0,25,30 : GOTO 1300
1470 PUT 12,10,10 : PRINT TAB(12);"CONSEGUIU. PARABENS" : PRINT CHR$(0);TAB(12);
"__________________";
1480 PUT 22,18,25,6 : PRINT "QUER JOGAR OUTRA VEZ ? ";CHR$(8);
1490 GET#5, A : IF A=83 PUT 31,7 : GOTO 1110
1500 PUT A,7,22,12,22 : END
2000 P$(18)="                " : P$(17)="                "
2010 P$(16)="                " : P$(15)="                "
2020 P$(14)="                " : P$(13)="                "
2030 P$(12)="                " : P$(11)="                "
2040 P$(10)="                " : P$( 9)="                "
2050 P$( 8)="                " : P$( 7)="                "
2060 P$( 6)="                " : P$( 5)="                "
2070 P$( 4)="                " : P$( 3)="                "
2080 P$( 2)="                " : P$( 1)="                "
2090 P$( 0)="                " : FOR I=1 TO 18: P(1,I)=19-I: NEXT I : PN(1)=18
2200 FOR I=1 TO 3
2210    X=12+19*(I-1)
2220    FOR J=1 TO 18
2230       Y=21-J
2240       PUT 22,X,Y : PRINT P$(P(I,J));
2250    NEXT J
2260 NEXT I
2270 RET

9999 END
```

CONVERSÃO DE PROGRAMAS DO ZX81 → ZX SPECTRUM

Autor: FERNANDO PRECES
Sacavém

PARTE I (continuação)

1.3 — PROGRAMA EM BASIC

O espaço reservado ao programa em Basic tem de ser **rigorosamente controlado,** porque a rotina em código máquina que se lhe segue contém endereços absolutos que não podem ser transviados.

Para esse control, deve utilizar a rotina (linhas 8999 e 9020) que lhe vai indicar o endereço em que termina cada linha Basic do programa. Esses endereços têm de ser coincidentes com os representados no quadro 3.1.

Sempre que um endereço verificado não condiga com o do quadro, terá de rever a respectiva linha, analisando bem as instruções e os espaços, pois forçosamente cometeu algum erro. Os mais vulgares nestes casos dão-se nas passagens ao INVERSE VÍDEO e na volta. Se isso acontecer, leia com atenção o manual da máquina.

PROGRAMA EM BASIC

```
   1 REM CONVERSOR
   2 GO SUB 4: RANDOMIZE USR 169
33: GO SUB 295: IF R$="C" THEN G
O TO 600
   3 GO TO 2
   4 LET PRAMT=FN S(23732): DIM
R$(1): DIM m$(128): LET J=0: LET
 E=0: LET N=0
   5 GO SUB 7000: LET I=PRAMT-FN
 S(23627)-800: DIM P$(1,I): LET
J=INT (I/256): POKE 17419,I-256*
J: POKE 17420,J: GO TO 100
 100 DEF FN s(x)=PEEK x+256*PEEK
 (x+1)
 101 DEF FN t(x)=PEEK (x+1)+256*
PEEK x
 230 POKE 23658,8
 270 LET M$="Introduzir o nome d
o programa do      ZX 81   seguido
de  ENTER": GO SUB 7000: INPUT n
$: LET m$="    Quer transformar t
ambem as        variaveis  ?    SI
M ou NAO": GO SUB 7000: LET i=11
8: LET j=0: INPUT "(S ou N) ";r$
: IF r$="S" THEN LET i=128: LET
j=1
 271 POKE 23490,i: POKE 16386,j
 280 LET m$=" Colocar a cassette
 no gravador,  ajustar devidamen
te o volume         e introduzir
 ENTER          ": GO SUB 7000: INPU
T r$
 285 LET m$="A pequena barra pre
ta (_) que semove na base do ecr
an indica queo programa foi loca
lizado e estaa ser transformado.
": GO SUB 7000: RETURN
 290 STOP
 295 LET E=PEEK 16408: GO TO 300
+E
 300 LET m$="  Leitura correcta
do programa          introduzir
 ENTER            para continuar a t
ransformacao.": GO SUB 7000: INP
UT R$: LET R$="C": RETURN
 555 GO SUB 7030: PRINT "Erro de
 leitura no. ";e: PRINT "Introdu
zir R para tentar de novoC para
 converter o que foi lido": INPU
T "(R ou C) ";r$: RETURN
 640 LET i=0: LET m$="Quer trans
formar os codigos das linhas RE
M  ?          SIM ou NAO": GO SUB 7
000: INPUT "(S ou N) ";r$: LET m
$="ESPERE": GO SUB 7000: IF r$="
N" THEN LET i=1
 645 POKE 16387,i
 650 LET i=USR 16695
 660 LET e=PEEK 16408: GO TO 700
+e
 700 LET m$="CONVERSAO COMPLETA
-Espere pelo  aparecimento do co
digo de erro  = B Integer out of
 range, = na  ultima linha e int
roduza LIST              seguido de E
NTER": GO SUB 7000: GO TO 9999
 954 GO SUB 7030: PRINT "Erro de
 conversao no. ";e: PRINT "Intro
duzir ENTER para colocar  os co
digos ja transformados .": INPUT
 r$: GO TO 9999
 955 GO SUB 7010: GO TO 600
7000 PRINT AT 16,0;m$: RETURN
7010 IF PEEK 16411=0 THEN GO SUB
 7030: PRINT FN t(16388);" REM..
...": GO TO 7012
7011 LET m$=CHR$ PEEK 16388+"$":
 GO SUB 7000
7012 PRINT AT 17,0;"  Converter
?          SIM ou NAO": INPUT "(S o
```

```
u N) ";r$: IF r$="N" THEN POKE 1
6410,1: RETURN
7013 POKE 16410,0: RETURN
7030 LET m$="": GO SUB 7000: PRI
NT AT 15,0: RETURN
8000 INVERSE USR 17230: STOP : R
EM Place code.
8999 INPUT "Endereco inicial ";x
9000 LET a=PEEK x
9010 IF a=13 THEN PRINT x;" fim
de linha"
9020 LET x=x+1: GO TO 9000
```

N.º da Linha	Endereço Fim de Linha	N.º da Linha	Endereço Fim de Linha
1	23769	555	25040
2	23826	640	25205
3	23839	645	25244
4	23927	650	25244
5	24064	660	25277
100	24106	700	25471
101	24148	954	25600
230	24173	955	25627
270	24426	7000	25655
271	24460	7010	25734
280	24588	7011	25772
285	24728	7012	25879
290	24734	7013	25906
295	24767	7030	25950
300	24904	8000	25984

Figura 3.1 — Quadro dos endereços

Uma vez conferido o programa, se tudo bater certo, apague as linhas 8999 a 9020.
Chame a linha 8000 e rectifique-a para número 9999.
Depois, com:

CLEAR : SAVE "CONVERSOR"

grave o programa.

(Continua no próximo n.º)

CAMPANHA NOVOS SÓCIOS

Durante os meses de JULHO e AGOSTO, o sócio que conseguir uma nova inscrição no CLUBE Z80 terá a oferta de uma CASSETE COM UM JOGO, à sua escolha.

FICHA DE CLÍNICA GERAL

Mais uma vez pedimos aos sócios sugestões para a criação de uma FICHA DE CLÍNICA GERAL, com vista à criação de um programa que use o "Microdrive" e que possa ser distribuído a todos os sócios que o solicitem.

LIVROS

Dados os vários pedidos que temos recebido para compra de livros já esgotados, passamos a referir os que ainda estão disponíveis para venda e as respectivas quantidades. Estes livros são reproduções dos originais e foram-nos vendidos pela "Landry" aos preços que fazemos aos sócios.

- The ZX Spectrum Explored (2 exemplares) — 420$00
- Advanced Graphics with the ZX Spectrum (4) — 650$00
- Programming your ZX Spectrum (3) — 600$00
- 60 Games & Applications for the ZX Spectrum (4) — 200$00
- 20 Best Programs for the ZX Spectrum (1) — 250$00
- 40 Best Machine Code Routines for the ZX Spectrum (2) — 360$00
- Master your ZX Microdrive (4) — 360$00
- The ZX Spectrum and how to get most from it (3) — 360$00
- Exploring Spectrum Basic (3) — 500$00
- Spectrum Spectacular (5) — 405$00
- The Spectrum Pocket Book (4) — 405$00
- Spectrum Microdrive Book (8) — 320$00
- Spectrum Machine Code Made Easy — Vol. I (4) — 540$00
- Spectrum Machine Code Made Easy — Vol. II (4) — 350$00
- ZX Spectrum Microdrive & Interface Manual — em Português (6) — 200$00
- 60 Programas for the Sinclair ZX Spectrum (4) — 700$00

VENDAS NA SEDE DO CLUBE OU À COBRANÇA

Nota: Ao encomendar algum livro ao CLUBE Z80, prevendo que ele já tenha esgotado, refira se, nesse caso, pretende que lhe enviemos fotocópias.

NOVOS PROGRAMAS SPECTRUM

JOGOS

CÓDIGO*/PREÇO

- ALIEN SWARM (16 K) — Com 3 naves espaciais, detenha os 50 invasores que, conforme vão sendo atacados, mais rápidos e agressivos se tornam. (Jogo idêntico ao "Invaders") **IP/400$00**
- CRAZY BALLOONS (16 K) — Controle um balão através de um labirinto, evitando colidir com as paredes. Quanto menos tempo gastar, maior será o bónus de pontuação. **IP/400$00**
- FIRE FLASH (16 K) — Use um laser e mísseis para defender uma base espacial do ataque inimigo. **IP/400$00**
- ICEBERG (16 K) — Tente salvar as equipas de investigadores do Oceano Ártico, controlando um quebra-gelos. Transporte cada equipa até ao porto, sem colidir com os "icebergs". **IP/400$00**
- JUNGLE TROUBLE (16 K) — Para chegar a casa, terá de atravessar um rio, de pedra em pedra, por causa do crocodilo. Entrando na floresta, tem de abater as árvores, prestando atenção aos macacos, e usando um machado. A seguir terá que transpor uma barragem de fogo, etc. **IP/400$00**
- KAMIKAZE (16 K) — Durante a sua viagem no espaço, tente destruir naves, radares ou posições inimigas. (Jogo idêntico ao "Ground Attack"). **IP/400$00**
- MOON ALERT (48 K) — "Space Fighter 7" é atacado por seres espaciais quando procura encontrar a sua base. Para além do terreno ser bastante acidentado, há também que ter em conta os disparos inimigos. **IP/400$00**
- MORRIS MEETS THE BIKERS (16 K) — Percorrendo vários cenários, ajude Morris a encontrar 10 moedas de ouro. **IP/400$00**
- TRANSVERSION (16 K) — O seu objectivo é percorrer toda a grelha galáctica, evitando os disparos dos guardas de todas as posições. No final de cada missão, haverá uma nova grelha. **IP/400$00**

UTILITÁRIOS

- MACRO CONSTRUCTION & ANIMATION (48 K) — Com este programa e a sua imaginação, construa os seus próprios jogos com música e animação viva. Pode escolher entre 30 figuras, movimentá-las e pô-las a falar. **I/700$00**
- THE QUILL (48 K) — Do género do anterior, este programa em C.M. permite-lhe fazer os seus próprios jogos de aventuras. Partindo de um menu, pode criar cerca de 200 localizações e preenchê-las com os objectos, problemas e descrições que quiser. **I/600$00**
- OMNICALC 2 (48 K) — Programa para simulação financeira; cálculo científico ou aplicações comerciais. Existe um quadro que pode ter até 99 colunas e 250 linhas, podendo ser preenchido com títulos, números ou fórmulas de cálculo.
  O computador irá calcular a matriz formada e pode recalcular em poucos segundos toda a matriz, quando se alteram quaisquer dados.
  Omnicalc 2 pode usar-se com **cassete** ou **microdrive** e pode trabalhar com **interface RS 232,** ou seja, permite saída para impressora de 80 colunas. (Microcassete) **I/2 400$00**
  (Cassete) **I/1 000$00**
- THE FOREST (48 K) — Simulação do desporto de orientação, para uso dos apaixonados de rallyes ou dos caminheiros. É ainda interessante para os que gostam de problemas de navegação e desejam treinar-se na técnica da orientação. Possui vistas de cortes no terreno, mapas com escala e desenhos do ambiente. É também útil para os professores de Geografia, particularmente na demonstração da relação entre os mapas e o meio físico. **I/600$00**
- ABERSOFT-FORTH (48 K) — Interpretador de linguagem FORTH. 50 vezes mais rápido que o BASIC. Possui comandos «gráficos». **I/600$00**
- MICRO-PROLOG (48 K) — Linguagem completamente diferente, usada nos círculos de Inteligência Artificial e Expert Systems. Possui vários programas explicativos e demonstrativos da potência desta linguagem. Pode «ensinar» o computador a reconhecer a descrição das «coisas» e desenhos e construir processos de aquisição de conhecimentos. **I/1 000$00**

* IP — Programa em inglês e instruções em português.
I — Programa e instruções em inglês.

C L U B E Z80

INSCRIÇÃO COMO ASSOCIADO

O **CLUBE Z80** está aberto a todos os utilizadores de microcomputadores.

A intenção de associar os entusiastas das micro-máquinas, é exclusivamente a de permitir:

1 — PUBLICAÇÃO DE UM JORNAL MENSAL, onde sejam publicados programas de uso geral ou específico como no caso da educação.

2 — PROMOVER TROCAS DE PROGRAMAS, e trocas de experiências; tanto no caso do Software (programação), como no caso do Hardware (electrónica).

3 — PROMOVER DESCONTOS NA AQUISIÇÃO DE PROGRAMAS.

4 — LANÇAR CURSOS DE PROGRAMAÇÃO EM BASIC — PASCAL OU OUTRAS LINGUAGENS E DIVULGAR O USO DE LINGUAGEM MÁQUINA.

NOME ....

IDADE .... COMPUTADOR TIPO ....

PROFISSÃO ....

ENDEREÇO ....

.... TELEF. ....

ASSINATURA ANUAL — Esc. 1 500$00 ☐

ASSINATURA SEMESTRAL — Esc. 750$00 ☐

CHEQUE OU VALE DO CORREIO

N.º ....

BANCO ....

DATA ..../......./.......

JÁ SÓCIO ☐

NOVO SÓCIO ☐ → A partir do mês de .... (inclusive)